



# **DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

José  
ETIS

Consultor: Manel Rella Ruiz

15/06/2014

UOC

## **RESUMEN**

El trabajo de fin de carrera (TFC) pertenece al área de la bases de datos relacionales. Para que el alumno demuestre las competencias adquiridas en este área se le presenta un problema vinculado al mundo real. Las competencias sobre este área son específicas de las asignaturas Bases de Datos I y II, pero también requieren el apoyo de otras. Mención especial merece Ingeniería de Software, esencial para un buen diseño de trabajo.

Las redes sociales son hoy día un ámbito consolidado para las relaciones interpersonales. Este tipo de relaciones pueden basarse en emociones, sentimientos, intereses comunes, actividades sociales...

Bajo este amplio campo han aparecido gran cantidad de aplicaciones que pretenden facilitar la actividad de los miembros de las redes sociales en cualquier ámbito.

El presente trabajo está asociado a esta vía y consiste en la creación de una BD para generar una aplicación que permita la gestión de un regalo grupal.

En cualquier momento un miembro de la red social puede generar un grupo, permitir adhesiones al mismo y reunir fondos para hacer un regalo a otro miembro.

El presente trabajo recoge las etapas de planificación, diseño, implementación y prueba de la BD.

Partiendo de los requisitos del cliente se presenta una modelización e implementación de la estructura de una BD, definiendo que datos se deben almacenar y como.

Profundizando también en los requerimientos del cliente se ofrecen también una serie de funcionalidades para permitir la inclusión o eliminación de datos en la BD.

La última parte del proyecto consiste en una serie de procedimientos que permiten calcular una serie de estadísticas y dar respuesta a una serie de consultas.

## ÍNDICE DE CONTENIDO

<b>RESUMEN.....</b>	<b>2</b>
<b>1. INTRODUCCIÓN .....</b>	<b>5</b>
1.1 JUSTIFICACIÓN DEL TFC Y CONTEXTO EN EL QUE SE DESARROLLA .....	5
1.2 OBJETIVOS DEL PROYECTO .....	5
1.2.1 OBJETIVOS GENERALES.....	5
1.2.2 OBJETIVOS ESPECÍFICOS.....	5
1.3 ENFOQUE Y MÉTODO A SEGUIR.....	6
1.3.1 ENFOQUE.....	6
1.3.2 MÉTODO A SEGUIR.....	7
1.4 PLANIFICACIÓN.....	7
1.4.1 FECHAS CLAVES .....	8
1.4.2 RELACIÓN DE TAREAS ESPECÍFICAS.....	8
1.4.3 DIAGRAMA DE GANTT.....	9
1.5 RESULTADOS ESPERADOS.....	11
1.5.1 ENTREGAS PARCIALES.....	11
1.5.2 PRODUCTO FINAL.....	11
1.6 BREVE DESCRIPCIÓN DE LOS OTROS CAPÍTULOS DE LA MEMORIA.....	12
<b>2. ANÁLISIS DE REQUISITOS.....</b>	<b>12</b>
2.1 MARCO DE REQUISITOS .....	12
2.2 REQUISITOS FUNCIONALES.....	13
<b>3. DISEÑO.....</b>	<b>16</b>
3.1 DISEÑO CONCEPTUAL .....	16
3.1.1 IDENTIFICACIÓN DE ENTIDADES Y ATRIBUTOS .....	17
3.1.2 ESQUEMA <i>ENTIDAD-RELACIÓN</i> .....	19
3.2 DISEÑO LÓGICO.....	21
3.3 DISEÑO FÍSICO .....	23
3.3.1 CREACIÓN DE LA BASE DE DATOS.....	23
3.3.2 CREACIÓN DEL TABLESPACE.....	23
3.3.3 CREACIÓN DEL USUARIO .....	23
3.3.4 CREACIÓN DE TABLAS .....	24
3.3.5 CREACIÓN DE ÍNDICES.....	24
<b>4. IMPLEMENTACIÓN DE FUNCIONALIDADES.....</b>	<b>24</b>
4.2 PROCEDIMIENTOS DE ALTA, BAJA Y MODIFICACIÓN.....	24
4.3 PROCEDIMIENTOS DE CONSULTA.....	32
4.4 PROCEDIMIENTOS DEL MÓDULO ESTADÍSTICO .....	34
<b>5. PLAN DE CONTINGENCIAS.....</b>	<b>40</b>
<b>6. PLAN DE PRUEBAS.....</b>	<b>41</b>
6.1 CARGA INICIAL DE DATOS .....	42
6.2 PRUEBAS DE ALTA, BAJA Y MODIFICACIÓN .....	42
6.3 PRUEBAS ESTADÍSTICAS.....	44
6.4 PRUEBAS CONSULTAS.....	45
6.4 COMPROBACIÓN LOGS.....	47
<b>7. RECURSOS NECESARIOS Y VALORACIÓN ECONÓMICA.....</b>	<b>48</b>
7.1 RECURSOS NECESARIOS .....	48
7.1.1 RECURSOS HUMANOS.....	48
7.1.2 RECURSOS <i>HARDWARE Y SOFTWARE</i> .....	48

<b>7.2 VALORACIÓN ECONÓMICA .....</b>	<b>49</b>
<b>8. CONCLUSIONES .....</b>	<b>50</b>
<b>8.1. CONCLUSIONES GENERALES .....</b>	<b>50</b>
<b>8.2. ALGUNAS PROPUESTAS DE MEJORA .....</b>	<b>50</b>
<b>9. BIBLIOGRAFÍA.....</b>	<b>51</b>
<b>10. ANEXO .....</b>	<b>52</b>

## **1. INTRODUCCIÓN**

### **1.1 JUSTIFICACIÓN DEL TFC Y CONTEXTO EN EL QUE SE DESARROLLA**

El TFC es una asignatura pensada para cursarse al final de la carrera y plantea un trabajo de síntesis sobre los estudios realizados durante los años previos.

Es obligada su correcta elaboración para poder obtener la titulación de la entidad docente, en nuestro caso UOC.

Las bases de datos relacionales son el ámbito de conocimientos en el que el alumno deberá centrarse para poder llevar a término el proyecto. Los conocimientos adquiridos en las asignaturas relacionadas no son excluyentes de utilizar sistemas de gestión de bases de datos no usados previamente. Se plantea, por tanto, el valor añadido de solicitar del alumno una ampliación de conocimientos.

Partiendo de un enunciado se deberá realizar un diseño de una base de datos que posteriormente se implementará para dar cumplimiento a los requerimientos que se plantean en el documento anteriormente citado.

### **1.2 OBJETIVOS DEL PROYECTO**

#### **1.2.1 OBJETIVOS GENERALES**

La lectura del plan docente, enunciado del TFC, documentación relativa a las buenas prácticas en la redacción de textos científicos y presentación de los mismos, unido a las nociones técnicas necesarias en relación a las bases de datos y técnicas de diseño, permiten inferir que el objetivo general es demostrar la capacitación del alumno en un área determinada de la informática.

Esta capacitación debe incluir todos los aspectos relativos a la asimilación de un enunciado de tipo generalista, que le permita la confección de un texto científico y producto que de resolución al problema planteado.

El problema se centra en el diseño de un BD para gestionar un sistema de regalos grupales entre sus miembros.

#### **1.2.2 OBJETIVOS ESPECÍFICOS**

Los requisitos son proporcionados por el cliente mediante el documento “Enunciado de TFC bases de datos relacionales”. Estos permiten determinar que los objetivos específicos del sistema de BD a desarrollar son:

- Almacenar los datos relacionados con los elementos que componen el sistema de gestión de regalos grupales:
  - Datos relacionados con la persona homenajeadada, los regalos seleccionados por el grupo o la dedicatoria general y el identificador de los miembros que forman el mismo, etc...
  - Datos relacionados con los regalos seleccionados por el grupo tales como código, categoría o estado de este.
  - Datos relacionados con las transacciones económicas tales como identificador de la persona que realiza el pago, identificador del grupo y regalo donde se ha realizado el pago, método de pago...
- Gestión de datos del sistema. Este aspecto está referido a los procedimientos que deben permitir operaciones tales como altas y bajas de grupos, participantes en grupos, homenajeados, regalos, etc...
- Disponer de un módulo estadístico que permita dar respuesta a una serie de consultas, partiendo de la base de que los datos deben estar siempre actualizados al momento.
  - Número de veces que se ha regalado un producto en un año concreto.
  - Número de regalos que se han adquirido en un año concreto.
  - Importe total de los regalos que se han adquirido en un año concreto.
  - Cantidad media de gasto en regalos de los miembros en un año concreto.
  - Porcentaje de regalos que han recaudado fondos suficientes frente a regalos abortados por fondos insuficientes en un año concreto.
  - Identificar que persona ha recibido más regalos en un año y mes concreto.
  - Identificar que persona ha gastado más dinero en un año y mes concreto.
  - Identificar la persona que ha participado en más regalos diferentes en un año y mes concreto.
  - Identificar el grupo que más ha gastado en un año y mes concreto.

### **1.3 ENFOQUE Y MÉTODO A SEGUIR**

En este apartado se determinan el enfoque y el método a seguir en el desarrollo del proyecto.

#### **1.3.1 ENFOQUE**

#### DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL

Definir la atención o interés hacia el problema que se expone en el enunciado permite establecer que método es el más apropiado para implementar el trabajo de fin de carrera.

En función de la lectura del enunciado, el plan docente, conocimientos previos de la materia a tratar y el desarrollo de productos similares, podemos inferir dos características básicas: el objetivo es claro y la solución es conocida.

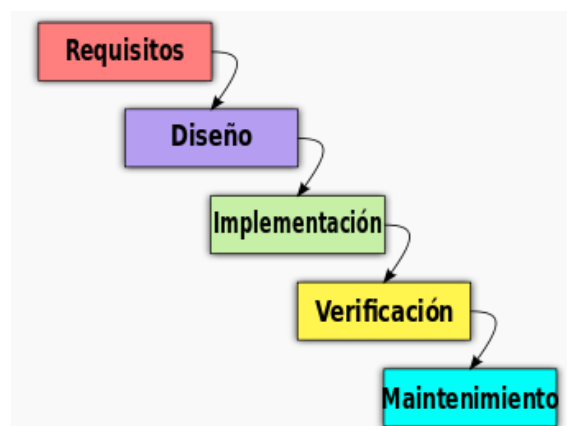
Por tanto, se puede recurrir a la clasificación de proyectos de *Wysocki* y determinar que se trata de un proyecto de grupo 1.

### 1.3.2 MÉTODO A SEGUIR

El método de desarrollo llamado *ciclo de vida clásico* o *en cascada* se ajusta de forma ideal a los proyectos de grupo 1. Este tipo de método es secuencial, sencillo de implementar y aplicar, pero presenta el inconveniente de ser bastante intolerante a cambios.

Se desarrolla en 5 etapas:

- Requisitos. Definen qué debe ser el producto a desarrollar
- Análisis y diseño. Define como debe de ser el producto.
- Implementación. Implica escribir código, manuales y generar el producto.
- Pruebas. Implica la verificación del producto desarrollado.
- Mantenimiento. Se pone el sistema a disposición de los usuarios y se corrigen los defectos detectados.



*Fases del ciclo de vida en cascada*

Conocido el método y las fases que lo constituyen, en el siguiente apartado se asociarán estas a las fechas claves que tiene el TFC.

### 1.4 PLANIFICACIÓN

La planificación del proyecto es el punto clave de este documento. Debe servir para acometer con solvencia el correcto desarrollo y la implementación del TFC.



En este apartado se establecen las pautas del método a seguir, en que etapas se dividirá la ejecución del proyecto y el tiempo a dedicar estimado para cada una de ellas.

La planificación abarca desde el 26 de febrero al 15 de junio de 2014, este plazo de tiempo acota 15 semanas de trabajo que a una media de entre 12 y 14 horas semanales ofrecen un saldo de entre 180 y 210 horas. Sería deseable que el desarrollo no superase el margen superior de horas, ya que, si se trata de la emulación del desarrollo de un producto, este desfase incurre de forma directa en pérdidas de beneficio para la entidad desarrolladora.

#### **1.4.1 FECHAS CLAVES**

Resulta pertinente, no obstante, no perder la perspectiva pedagógica de esta asignatura. En base a ello se pueden establecer unas tareas básicas asociadas a las fechas claves que permiten una primera aproximación a las labores a realizar.

Entrega	Tarea asociada	Fecha
PAC1	Plan de trabajo	16-03-2014
PAC2	Diseño	13-04-2014
PAC3	Implementación y pruebas	11-05-2014
Entrega final	Producto, memoria y presentación	15-06-2014
Tribunal virtual	Defensa del proyecto	25-06-2014

#### **1.4.2 RELACIÓN DE TAREAS ESPECÍFICAS**

En este apartado se detallan cuales son de forma específica las tareas que engloban cada una de las tareas generales apuntadas en el apartado anterior.

- Plan de trabajo
  - Lectura de enunciado y plan docente
  - Identificación de tareas básicas a realizar
  - Archivo estructurado de documentación de referencia básica: apuntes, manuales, ejemplos...
  - Instalación y puesta en marcha de software para la ejecución del TFC: máquina virtual, Oracle, SQL Developer, Word...
  - Redacción, revisión y entrega de PAC1
- Análisis y Diseño
  - Revisión de requisitos
    - Identificación inicial de requisitos
    - Resolución y aclaración de incertidumbres sobre requisitos
  - Identificación precisa de requisitos



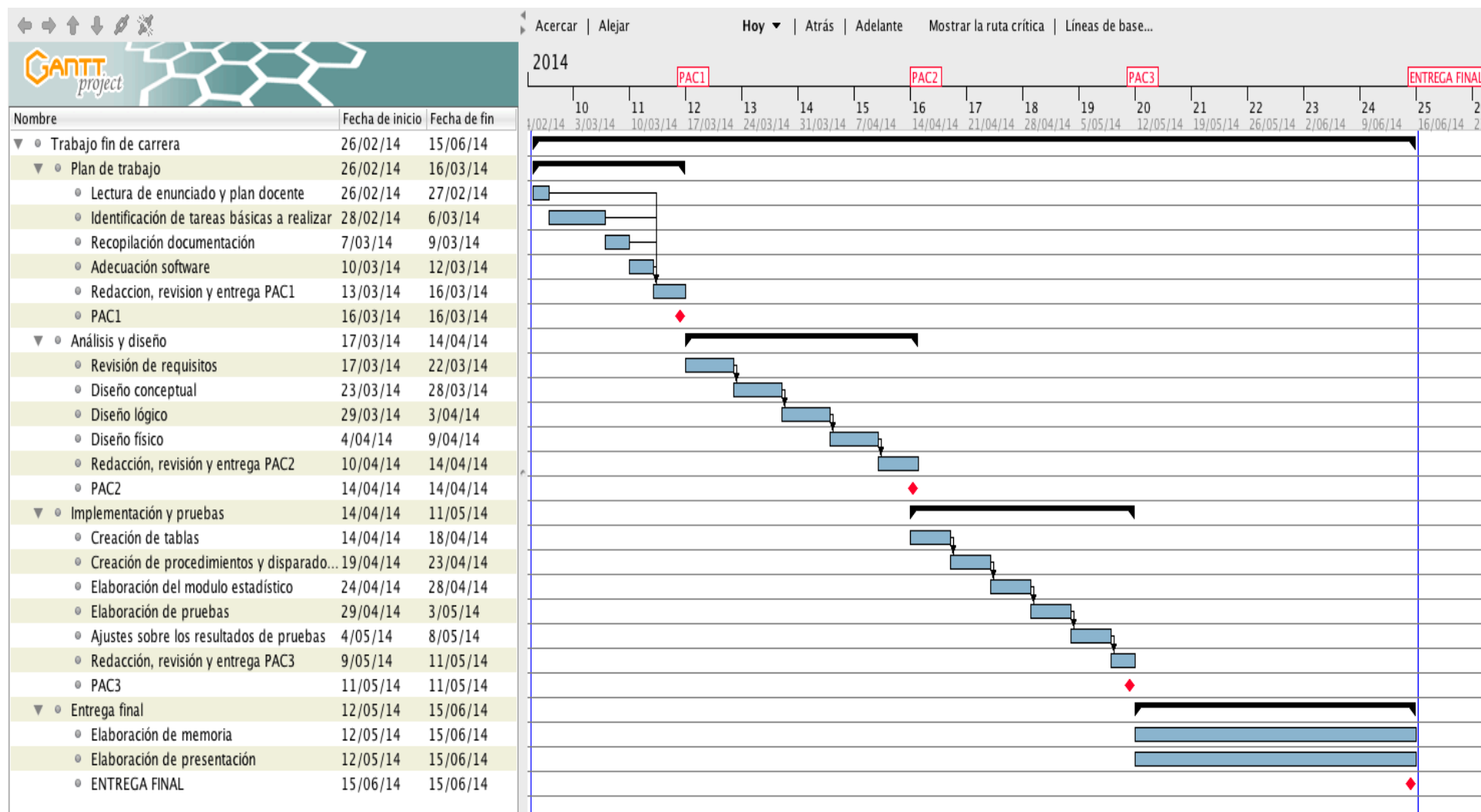
**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

- Diseño conceptual
  - Diseño lógico
  - Diseño físico
  - Redacción, revisión y entrega PAC2
- Implementación y pruebas
    - Creación de tablas
    - Creación de procedimientos y disparadores
    - Elaboración del modulo estadístico (similar a Datawarehouse)
    - Elaboración de pruebas
    - Ajustes sobre los resultados de pruebas
    - Redacción, revisión y entrega PAC3
  - Entrega final
    - Elaboración de memoria
    - Elaboración de presentación

#### **1.4.3 DIAGRAMA DE GANTT**

Definidas las tareas sólo cabe, por último, asignar una temporización a cada una de ellas mediante un diagrama de Gantt.

DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL



## 1.5 RESULTADOS ESPERADOS

En este apartado se detallan en que consistirá el producto obtenido del seguimiento de la planificación.

### 1.5.1 ENTREGAS PARCIALES

Tal como se comentaba anteriormente, el TFC tiene un aspecto didáctico porque está enfocado a conseguir que el alumno aplique, desarrolle y amplíe una vertiente de los conocimientos tratados en la carrera.

Para cubrir esta faceta se realizan una serie de entregas parciales que se han mencionado en el apartado dedicado a la planificación.

La serie de entregas parciales son:

- PAC1. Recoge todos los aspectos relativos a la planificación del proyecto y es también un documento introductorio a los objetivos del mismo.
- PAC2. Debe contener todo lo relacionado con la metodología de análisis con el objetivo de obtener las especificaciones de los requerimientos y permitir un diseño solvente de la BD.
- PAC3. Esta entrega se centra en la implementación de la BD y contiene todos los scripts necesarios para su funcionamiento. También incluye la etapa de pruebas y rectificación para completar la implementación.

### 1.5.2 PRODUCTO FINAL

El tratamiento de la documentación parcial debe ser el embrión del producto final, objetivo último del TFC.

El producto final constará de:

- Producto. Esta parte incluye todos los scripts necesarios para que la base de datos quede totalmente operativa y cumpla de forma estricta los requerimientos del cliente.
- Memoria. Es un documento que detalla todas las fases que incluye el trabajo de desarrollo de la BD. Desde el plan de trabajo hasta todos los detalles relacionados con todas las fases de desarrollo del proyecto.
- Presentación virtual. Estará formada por una serie de diapositivas que permitirán conocer el proyecto de una forma concisa y dinámica.

## **1.6 BREVE DESCRIPCIÓN DE LOS OTROS CAPÍTULOS DE LA MEMORIA**

Los siguientes capítulos documentan todo el proceso acometido para la creación de la base de datos y funcionalidades requeridas.

El capítulo 2 hace referencia al análisis de todos los requisitos que deben ser tomados en cuenta para realizar un buen diseño. Este último aspecto se acomete en el capítulo 3.

La implementación de todos los procesos relativos a las altas, bajas y modificaciones, así como los destinados a funciones estadísticas o consultas son especificados en el capítulo 4.

Especial atención requiere el testeo y prueba del trabajo realizado que se puede consultar en el capítulo 6.

El resto de apartados están dedicados a la documentación correspondiente a aspectos tan dispares como el plan de contingencias (capítulo 5), valoración económica del TFC (capítulo 7) o conclusiones (capítulo 8).

## **2. ANÁLISIS DE REQUISITOS**

En este capítulo se realiza un análisis detallado de los requisitos expuesto por el cliente.

Se considera requisito una característica observable del sistema que satisface una necesidad o expresa una restricción que concierne al sistema que se está desarrollando.

El análisis, también, irá acompañado de comentarios con el objetivo de despejar dudas y poner en común los requerimientos del cliente y las tareas finales a realizar.

### **2.1 MARCO DE REQUISITOS**

Una importante red social ha solicitado a algunos estudiantes de la UOC el diseño de una base de datos para gestionar un sistema de regalos grupales entre sus miembros.

A nivel general los miembros registrados en el red social pueden registrar cuales son los eventos susceptibles de homenaje (cumpleaños, boda, onomástica, etc...) y cuales son sus regalos preferidos.

La base de datos debe permitir formalizar un grupo que tendrá como objetivo reunir fondos para regalar un artículo a uno de sus miembros.

A nivel específico el objetivo la base de datos también debe permitir guardar toda una serie de información relacionada con la situación antes planteada, como por ejemplo: estado del regalo, transacciones realizados, estado de la reunión de fondos...

Todo este bloque de datos también debe dar soporte a una serie de consultas y permitir la obtención de información estadística.

El marco de colaboración con la UOC se centra únicamente en el diseño de la BD, ya que la implementación de alto nivel será responsabilidad de la propia red social, por tanto, no es necesario implementar ningún tipo de interficie gráfica.

## 2.2 REQUISITOS FUNCIONALES

R1	<p><b>Descripción</b></p> <p>El modelo debe permitir guardar los datos básicos de los "grupos de participantes en los regalos", estos datos serían como mínimo:</p> <ul style="list-style-type: none"> <li>- Identificador de la persona homenajeadada destinataria final del regalo .</li> <li>- Regalos seleccionados por el grupo de entre la lista de deseos de la persona homenajeadada.</li> <li>- Dedicatoria general de todo el grupo.</li> <li>- Los identificadores de los miembros que forman el grupo (en este caso serían los correos electrónicos con los que se han registrado en la red social, sus datos concretos ya estarán dados de alta en la red social), así como : <ul style="list-style-type: none"> <li>o Su dedicatoria personal a la persona homenajeadada (de forma opcional).</li> <li>o La cantidad económica con la que ha participado cada miembro en cada regalo. Cada participante será libre de participar en el regalo que quiera con el importe que considere más adecuado .</li> </ul> </li> <li>- Fecha límite de recogida de fondos .</li> <li>- Identificador de la persona receptora del regalo (que puede coincidir con la persona homenajeadada o bien ser una persona que recibirá el regalo y que hará después la entrega en persona) .</li> <li>- Dirección postal de envío de regalos.</li> <li>- Fecha límite para la recepción de los regalos .</li> <li>- Booleano que indicará si el importe de cada participación será público dentro del grupo o si por el contrario sólo se mostrará que un amigo ha participado en el regalo .</li> <li>- Etc .</li> </ul>
R2	<p><b>Descripción</b></p> <p>El modelo también deberá permitir guardar los datos asociados a los regalos elegidos por un grupo :</p> <ul style="list-style-type: none"> <li>- Los regalos se identificarán por su código de producto y estarán asociados a un proveedor concreto .</li> </ul>

	<ul style="list-style-type: none"> <li>- Los regalos también tendrán una categoría asociada . Las categorías estarán codificadas y se podrán ampliar en un futuro.</li> <li>- Cada regalo tendrá también un coste económico asociado y un tipo de IVA concreto : 4% , 10% , 21%, etc.</li> <li>- Durante el ciclo de vida de los regalos , estos se encontrarán en diferentes estados : <ul style="list-style-type: none"> <li>o AFEGIT_A_UN_GRUP : este será el estado inicial una vez un regalo añada a un grupo de participantes.</li> <li>o ELIMINADO: en caso de que el grupo de participantes decida eliminar el regalo.</li> <li>o NO_SUFICIENTS_FONS : en caso de que se llegue a la fecha límite de recogida de fondos y no se haya llegado al importe necesario.</li> <li>o PAGAMENT_COMPLETAT: en caso de que se llegue al importe necesario entre todas las aportaciones.</li> <li>o ENVIAT</li> <li>o REBUT</li> </ul> </li> <li>- Cuando elimine un regalo los fondos que tuviera ya asignados se devolverán automáticamente a los participantes para dar la oportunidad de destinarlos a otro regalo .</li> </ul>
--	--

R3	<p><b>Descripción</b></p> <p>Todas las transacciones económicas (pagos y devoluciones) se deberán registrar de forma automática en una tabla de transacciones para posteriores auditorías, esto incluye guardar como mínimo:</p> <ul style="list-style-type: none"> <li>- Identificador de la persona que realiza el pago o recibe la devolución.</li> <li>- Identificador del grupo y del regalo asociado donde se ha realizado el pago.</li> <li>- El modo de pago que se ha empleado y los datos asociados a este: <ul style="list-style-type: none"> <li>o Número de cuenta bancaria</li> <li>o Número tarjeta de crédito</li> <li>o Cuenta paypal</li> <li>o etc.</li> </ul> </li> <li>- Cantidad económica de la transacción: positiva en caso de que sea un pago o negativa en caso de ser una devolución.</li> </ul>
----	--

R4	<p><b>Descripción</b></p> <p>El sistema también deberá contemplar tablas para guardar los catálogos de regalos de cada proveedor.</p>
----	---

R5	<p><b>Descripción</b></p>
----	---------------------------

	<p>La aplicación deberá disponer, como mínimo, de las siguientes funcionalidades, cumpliendo con los requisitos expresados previamente:</p> <ul style="list-style-type: none"> <li>• Procedimientos de consulta que permitan obtener la información siguiente : <ul style="list-style-type: none"> <li>a. Dado un miembro de la red social: el listado de sus regalos deseados; incluyendo el precio y el proveedor asociado, así como la fecha de alta del deseo.</li> <li>b. El listado de todos los grupos de participantes que todavía tienen algún regalo pendiente, es decir, que todavía no han recogido los fondos necesarios para que todos sus regalos se hayan podido adquirir. Indicando también la suma del importe total de todos los regalos, el importe que aún resta conseguir y el porcentaje respecto al importe total que se ha alcanzado .</li> <li>c. Dada una categoría: el listado de todos los regalos asociados, incluyendo los proveedores que los proporcionan.</li> <li>d. Dado un usuario de la red social: el listado detallado de todos los regalos que ha recibido.</li> <li>e. El listado de las personas que se encuentran por encima de la media a nivel del dinero que han destinado a regalos grupales.</li> <li>f. Dado un año: el listado de las 10 ciudades donde más regalos han recibido las personas homenajeadas durante el año indicado , así como el valor económico total correspondiente a la suma de todos los regalos en cada una de estas 10 ciudades.</li> </ul> </li> <li>• Se valorará la implementación de otros procedimientos o funcionalidades que puedan ser de utilidad, siempre y cuando estén documentadas y consensuadas con el consultor.</li> </ul>
--	---

R7	Descripción
	<p>Módulo estadístico, una parte muy importante del trabajo es la implementación de un módulo estadístico que debe alimentarse a partir de los procedimientos que implementen las funcionalidades mencionadas, a fin de ofrecer los siguientes datos en tiempo constante 1, es decir, hacer 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).</p>



	<p>El módulo estadístico deberá dar respuesta a las siguientes consultas:</p> <ol style="list-style-type: none"> <li>1. Dado un producto y un año concreto: el número de veces que se ha regalado este a diferentes personas. No hay que contemplar los regalos que nunca se han regalado .</li> <li>2. Dado un año concreto: el número de regalos totales que se han adquirido utilizando nuestra aplicación.</li> <li>3. Dado un año concreto: el importe total de los regalos que se han adquirido utilizando nuestra aplicación.</li> <li>4. Dado un año concreto: el porcentaje de regalos que han alcanzado los fondos respecto a los que han quedado en el estado " NO_SUFICIENTS_FONS".</li> <li>5. Dado un año concreto: la cantidad media de gasto en regalos de los miembros de la red social.</li> <li>6. Dado un año y un mes concreto: el identificador de la persona que ha recibido más regalos.</li> <li>7. Dado un año y un mes concreto: el identificador de la persona que más gasto en regalos ha realizado.</li> <li>8. Dado un año y un mes concreto: el identificador de la persona que ha participado en más regalos diferentes.</li> <li>9. Dado un año y un mes concreto: el identificador del grupo que más gasto en regalos ha realizado.</li> </ol> <p>Las respuestas del módulo estadístico deben ser inmediatas y éste debe estar siempre actualizado con la última información de la BD , es decir , NO se pueden utilizar jobs que se ejecuten por la noche y que rellenen los datos estadísticos (aunque éste sea un procedimiento habitual en el ámbito de las BD para data warehouse).</p>
--	---

### 3. DISEÑO

El diseño de la base de datos se descompone en tres etapas: *diseño conceptual*, *diseño lógico* y *diseño físico*.

#### 3.1 DISEÑO CONCEPTUAL

En esta etapa el diseño conceptual está orientada a centrarse únicamente en el problema de la estructura de la información. Es, por tanto, un diseño que se mantiene totalmente independiente del tipo de base de datos que se utilizará y la tecnología que se empleará en su implementación.

### 3.1.1 IDENTIFICACIÓN DE ENTIDADES Y ATRIBUTOS

Los atributos de las entidades que figuran en el diagrama son las siguientes (las claves primarias se han subrayado):

#### **CATEGORIA**

código, nombre-categoría, descripción

- *nombre-categoría* indica que tipo de artículo es, será necesario para posteriores catalogaciones de artículos.

#### **PROVEEDOR**

código, nombre-proveedor

#### **ARTÍCULO**

código, precio, descripción, IVA

- *descripción* tiene como función definir que es el artículo.
- *IVA* determina la carga impositiva del artículo que según la legislación vigente puede ser del 4%, 10% o el 21%.

#### **MIEMBRO**

código, e-mail, nombre, apellido1, apellido2, dirección, ciudad, país

- Esta entidad guarda un registro de los miembros de la red social. Se definen unos atributos mínimos que permiten completar el registro.
- *e-mail* también puede ser clave de la entidad, es indispensable que dos miembros de la red no se registren con el mismo correo.
- *dirección* guarda la dirección de residencia del miembro de la red.
- *ciudad* guarda la ciudad de residencia del miembro de la red.
- *país* guarda el país de residencia del miembro de la red.

#### **GRUPO**

código, *dedicatoria\_grupal*

- Esta entidad guarda el grupo que permitirá adhesiones para reunir fondos.
- *Dedicatoria\_grupal* es un mensaje colectivo del grupo al homenajeado.

#### **TRANSACCIÓN**

código, cantidad, tipo, *traza\_abono*, fecha, visible

- Esta entidad debe guardar el registro de las transacciones económicas, a pesar de que estamos en una etapa temprana del diseño es conveniente reseñar que aportará mayor operatividad en futuras fases del mismo.
- *tipo* define si se efectúa un pago un abono.
- *traza\_abono* guarda la procedencia de la transacción en un abono, sirve

para evitar que una cantidad sea abonada dos veces.

- *visible* es un requerimiento del cliente que indica sí el miembro desea que se vea su cantidad aportada o no.

#### **MÉTODO**

código, nombre-método

#### **EST1\_VECES\_REGALADO**

código, artículo, año, n\_veces

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.1

#### **EST2\_TOTAL\_REGALOS**

código, año, total\_regalos

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.2

#### **EST3\_IMPORTE\_TOTAL**

código, año, importe

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.3

#### **EST4\_PORCENTAJE\_COMPLETADO**

código, año, porcentaje

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.4

#### **EST5\_MEDIA\_GASTO**

código, año, media

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.5

#### **EST6\_MIEMBRO\_MAS\_HOMENAJEDO**

código, año, mes, miembro, n\_veces

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.6

#### **EST7\_MIEMBRO\_MAS\_GASTADOR**

código, año, mes, miembro, cantidad

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento

al requisito R7.7

### **EST8\_MIEMBRO\_MAS\_PARTICIPATIVO**

código, año, mes, miembro, n\_veces

- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.8

### **EST9\_GRUPO\_MAS GASTADOR**

código, año, mes, grupo, total

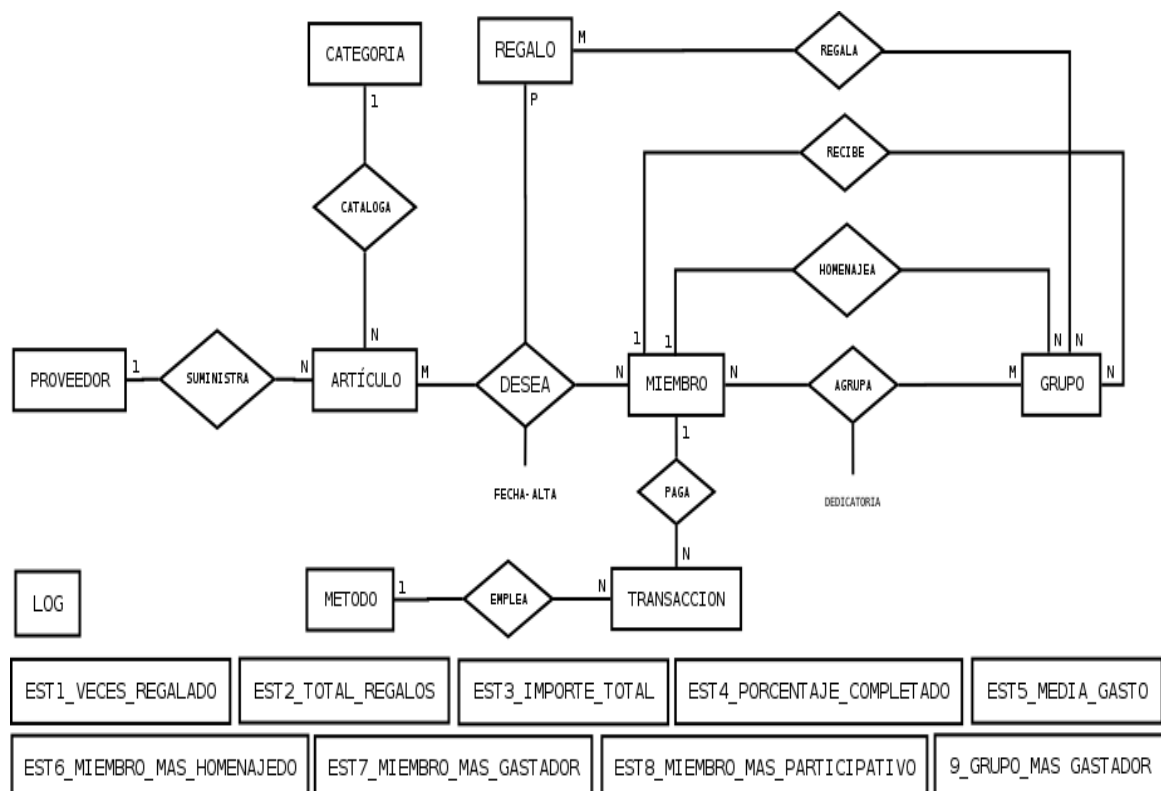
- Entidad relacionada con el módulo estadístico que debe dar cumplimiento al requisito R7.9

### **LOGS**

código, Procedimiento, Parámetro\_entrada, Parámetro\_salida

- Entidad destinada al almacenamiento de la información *log* relacionada con la ejecución de los procedimientos almacenados.

### **3.1.2 ESQUEMA ENTIDAD-RELACIÓN**



### **3.1.2.1 ARGUMENTACIÓN DEL ESQUEMA ENTIDAD-RELACIÓN**

- La relación entre ARTÍCULO y PROVEEDOR es 1:N porqué se considera que un proveedor suministra más de un artículo, pero un mismo artículo sólo puede ser suministrado por un proveedor. Este último punto es una restricción derivada de los requisitos del cliente.
- La relación entre ARTÍCULO y CATEGORIA es 1:N porqué se considera que un artículo únicamente puede pertenecer a una categoría, pero una categoría recoge un número indeterminado de artículos.
- La relación entre ARTÍCULO y MIEMBRO es M:N porqué se considera que una miembro puede seleccionar varios artículos y un artículo puede ser seleccionado por varios miembros.
- La relación entre MÉTODO y TRANSACCIÓN es 1:N porqué se considera que un método de pago puede ser empleado por varias transacciones, pero una transacción sólo puede emplear un método de pago.
- La relación entre MIEMBRO y TRANSACCIÓN es 1:N porqué se considera que un miembro puede realizar varias transacciones, pero una transacción sólo se puede asociar a un miembro.
- La relación entre MIEMBRO y GRUPO, utilizando la interrelación AGRUPA, es M:N porqué se considera que un grupo puede ser formado por varios miembros y un miembro de la red puede formar parte de varios grupos.
- La relación entre MIEMBRO y GRUPO, utilizando la interrelación HOMENAJEA, es 1:N porqué se considera que un miembro puede ser homenajeado por varios grupos, pero un grupo sólo puede tener un homenajeado.
- La relación entre MIEMBRO y GRUPO, utilizando la interrelación RECIBE, es 1:N porqué se considera que un miembro puede ser receptor en varios grupos, pero un grupo sólo puede tener un receptor.
- La relación entre GRUPO y REGALO es M:N porque se considera que un regalo puede ser seleccionado por varios grupos y un grupo puede seleccionar varios regalos.

### 3.2 DISEÑO LÓGICO

Partiendo del diseño conceptual se inician una serie de transformaciones sobre este, de modo que quede adaptado al SGBD que se desea implementar. El diseño lógico, para este caso, se basará en un SGBD relacional y el resultado obtenido serán una serie de relaciones entre sus atributos, claves principales y claves foráneas.

**CATEGORÍA**

id\_categoria, nombre, descripción

**PROVEEDOR**

id\_proveedor, nombre

**ARTÍCULO**

id\_artículo, categoría, proveedor, IVA, precio, descripción

Donde {categoría} es clave foránea en CATEGORÍA (id\_categoria)

Donde {proveedor} es clave foránea en PROVEEDOR (id\_proveedor)

**MIEMBRO**

id\_miembro, e-mail, evento, nombre, apellido1, apellido2, dirección, ciudad, país

**DESEA**

id\_deseo, homenajeado, artículo, fecha-alta

Donde {homenajeado} es clave foránea en MIEMBRO (id\_miembro)

Donde {regalo} es clave foránea en ARTÍCULO (id\_artículo)

**GRUPO**

id\_grupo, homenajeado, receptor, dedicatoria-grupal

Donde {homenajeado} es clave foránea en MIEMBRO (id\_miembro)

Donde {receptor} es clave foránea en MIEMBRO (id\_miembro)

**AGRUPA**

grupo, integrante, dedicatoria\_personal

Donde {grupo} es clave foránea en GRUPO (id\_grupo)

Donde {integrante} es clave foránea en MIEMBRO (e-mail)

**REGALA**

id\_regalo, grupo, deseo, fecha\_inicio\_recaudación, fecha\_final\_recaudación, estado\_regalo

Donde {grupo} es clave foránea en GRUPO (id\_grupo)

Donde {deseo} es clave foránea en DESEA (id\_deseo)

#### **MÉTODO**

id\_método, método

#### **TRANSACCIÓN**

id\_transacción, regalo, aportador, cantidad, tipo, traza\_abono, fecha, forma\_pago, visible

Donde {regalo} es clave foránea en REGALA(id\_regalo)

Donde {aportador} es clave foránea en MIEMBRO (id\_miembro)

Donde {forma\_pago} es clave foránea en MÉTODO (id\_método)

#### **EST1\_VECES\_REGALADO**

artículo, año, n\_veces

#### **EST2\_TOTAL\_REGALOS**

año, total\_regalos

#### **EST3\_IMPORTE\_TOTAL**

año, importe

#### **EST4\_IMPORTE\_COMPLETADO**

año, porcentaje

#### **EST5\_MEDIA\_GASTO**

año, media

#### **EST6\_MIEMBRO\_MAS\_HOMENAJEDO**

año, mes, miembro

#### **EST7\_MIEMBRO\_MAS\_GASTADOR**

año, mes, miembro

#### **EST8\_MIEMBRO\_MAS\_PARTICIPATIVO**

año, mes, miembro

#### **EST9\_GRUPO\_MAS GASTADOR**

año, mes, grupo

#### **LOGS**

Id\_Log, Procedimiento, Parámetro\_entrada, Parámetro\_salida



### 3.3 DISEÑO FÍSICO

En esta etapa se transforma la estructura obtenida en la etapa de diseño lógico, con el objetivo de adaptarla a los aspectos particulares del SGBD utilizado. En este caso se empleará ORACLE.

A continuación se detalla los pasos que debe llevar a la implementación de la base de datos que posteriormente se utilizará.

#### 3.3.1 CREACIÓN DE LA BASE DE DATOS

De forma básica podemos crear un BD utilizando Oracle:

- a) Utilizando la aplicación *Database Configuration Assistant (DBCA)*.
- b) Utilizar la sentencia *Create Database* y crearla de forma manual.

No obstante en este caso utilizaremos la base de datos que se crea por defecto durante la instalación de Oracle. Se asume, por tanto, que el cliente dispone de una base de datos operativa y configurada.

#### 3.3.2 CREACIÓN DEL TABLESPACE

A continuación creamos el *tablespace* que es una unidad lógica de almacenamiento dentro de una base de datos Oracle.

Este elemento es un puente entre el sistema de ficheros y la bases de datos.

Cada tabla o índice que se cree posteriormente debe ser asignado a un *tablespace* determinado.

En nuestro caso utilizaremos la siguiente instrucción:

```
CREATE TABLESPACE BD_tablas  
DATAFILE 'C:\oraclexe\app\oracle\oradata\XE\BD_tablas.dbf' SIZE 50M  
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
```

También es viable, en función de la orientación que adquieran algunos aspectos de la base de datos, generar un *tablespace* destinado a índices. De esta forma es posible incrementar la eficiencia.

```
CREATE TABLESPACE BD_indices  
DATAFILE 'C:\oraclexe\app\oracle\oradata\XE\BD_indices.dbf' SIZE 50M  
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
```

#### 3.3.3 CREACIÓN DEL USUARIO

Es necesario crear un usuario con un nombre de acceso a la base de datos. Este usuario también lleva asociado normalmente una clave (password). En función de

los permisos que tenga asignados podrá realizar un determinado número de tareas.

Para nuestro caso el usuario es TFC con la clave TFC. Es necesario indicar que *tablespace* puede utilizar, en nuestro caso BD\_tablas, otorgar las cuotas de uso en los *tablespace* y, por último los permisos.

Utilizaremos la siguiente sentencia:

```
CREATE USER TFC
IDENTIFIED BY TFC
DEFAULT TABLESPACE BD_TABLAS
QUOTA UNLIMITED ON BD_TABLAS
QUOTA UNLIMITED ON BD_INDICES
```

```
TEMPORARY TABLESPACE TEMP;
```

```
GRANT CREATE SESSION, CREATE TABLE, CREATE SEQUENCE, CREATE TRIGGER, CREATE
TYPE, CREATE PROCEDURE
TO TFC;
```

### 3.3.4 CREACIÓN DE TABLAS

El siguiente paso consiste en la creación de las tablas que forman la base de datos. Se parte de la etapas de diseño conceptual y lógico y se elaboran un diseño físico que desemboque en la creación de las mismas.

En el apartado 10 de anexo se encuentra un desglose de todas las tablas y sus correspondientes especificaciones.

### 3.3.5 CREACIÓN DE ÍNDICES

Aunque Oracle crea por defecto un sistema de índices, en la base de datos del sistema se introducen en ocasiones la creación de índices adicionales con el objetivo de mejorar la eficiencia.

## 4. IMPLEMENTACIÓN DE FUNCIONALIDADES

A continuación se describen los procedimientos implementados según los requerimientos funcionales. Cada tabla especifica de forma esquemática el funcionamiento de una funcionalidad permitiendo saber de forma exacta su funcionamiento.

### 4.2 PROCEDIMIENTOS DE ALTA, BAJA Y MODIFICACIÓN

SP_ALTA_GRUPO	
Función	
- Da de alta un grupo	
Parámetros de entrada	
-	p_dedicatoria_grupal: dedicatoria común de todo el grupo.

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>- p_homenajead: código de un miembro que recibe el regalo.</li> <li>- p_receptor: código de un miembro que entrega el regalo.</li> </ul>
<b>Parámetros de salida</b>
<ul style="list-style-type: none"> <li>- RSP VARCHAR2</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el homenajead tiene uno o más deseos.</li> <li>- Comprueba que el receptor es miembro de la red.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_ALTA_GRUPO"</li> <li>- Ejecución fracasada: ERROR SP_ALTA_GRUPO: Se debe indicar una dedicatoria grupal. ERROR SP_ALTA_GRUPO: El homenajead no desea ningún regalo. ERROR SP_ALTA_GRUPO: El receptor no es miembro de la red social. ERROR SP_ALTA_GRUPO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Sólo miembros de la red que tiene deseos pueden ser homenajeados.</li> <li>- Es posible no indicar en el procedimiento de alta el homenajead. Se puede utilizar posteriormente para esta función el procedimiento correspondiente.</li> <li>- Es posible no indicar el receptor.</li> <li>- Si no se indica el receptor, pero si el homenajead, el receptor es por defecto el homenajead.</li> </ul>

<b>SP_BAJA_GRUPO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Da de baja un grupo.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_id_grupo: código único del grupo.</li> </ul>
<b>Parámetros de salida</b>
<ul style="list-style-type: none"> <li>- RSP VARCHAR2</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el grupo no tiene miembros dados de alta.</li> <li>- Comprueba que el grupo no tiene regalos asignados.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_BAJA_GRUPO"</li> <li>- Ejecución fracasada: ERROR SP_BAJA_GRUPO: El campo grupo no puede ser nulo. ERROR SP_BAJA_GRUPO: El campo grupo tiene participantes, deben darse de baja previamente. ERROR SP_BAJA_GRUPO: El campo grupo tiene regalos, deben darse de baja previamente. ERROR SP_BAJA_GRUPO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<p>El procedimiento de baja de un regalo presenta dos limitaciones:</p> <ul style="list-style-type: none"> <li>a) No se puede dar de baja un grupo que tiene participantes. Estos se pueden dar de baja con su correspondiente método.</li> </ul>

- b) No se puede dar de baja un grupo con regalos añadidos. Estos se pueden dar de baja con su correspondiente método.

<b>SP_ALTA_REGALO</b>	
<b>Función</b>	
- Da de alta un regalo dentro de un grupo.	
<b>Parámetros de entrada</b>	
<ul style="list-style-type: none"> <li>- p_grupo: id_del grupo.</li> <li>- p_deseo: id_del deseo del homenajeado del grupo.</li> <li>- p_fecha_final_recaudación: fecha límite para realizar aportaciones.</li> </ul>	
<b>Parámetros de salida</b>	
- RSP VARCHAR2	
<b>Comprobaciones</b>	
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que existe el grupo.</li> <li>- Comprueba que existe el deseo.</li> <li>- Comprueba que el homenajeado está dado de alta en el grupo.</li> <li>- Comprueba que el deseo no se adjudica al grupo de forma repetida.</li> <li>- Comprueba que el homenajeado del grupo y el solicitante del deseo son la misma persona.</li> </ul>	
<b>Salida</b>	
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_ALTA_REGALO"</li> <li>- Ejecución fracasada: ERROR SP_ALTA_REGALO: El campo grupo no puede ser nulo. ERROR SP_ALTA_REGALO: El campo deseo no puede ser nulo. ERROR SP_ALTA_REGALO: El campo fecha_final_recaudación no puede ser nulo. ERROR SP_ALTA_REGALO: El grupo no esta disponible. ERROR SP_ALTA_REGALO: El deseo no esta disponible. ERROR SP_ALTA_REGALO: El grupo no tiene homenajeado. Debe darse de alta previamente. ERROR SP_ALTA_REGALO: Ya existe ese regalo para ese grupo. ERROR SP_ALTA_REGALO: {Tipo de error} en cualquier otro caso.</li> </ul>	
<b>Consideraciones</b>	
- Regalo es la asociación de un deseo a un grupo.	

<b>SP_BAJA_REGALO</b>	
<b>Función</b>	
- Da de baja un regalo dentro de un grupo.	
<b>Parámetros de entrada</b>	
- p_regalo: id_regalo a eliminar	
<b>Parámetros de salida</b>	
- RSP VARCHAR2	
<b>Comprobaciones</b>	
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo</li> <li>- Comprueba que el regalo existe</li> <li>- Comprueba que el regalo se encuentra en estado "AFEGIT_A_UN_GRUP".</li> </ul>	
<b>Salida</b>	
- Ejecución finalizada con éxito:	

<p>“OK_BAJA_REGALO”</p> <ul style="list-style-type: none"> <li>- Ejecución fracasada: ERROR SP_BAJA_REGALO: El campo regalo no puede ser nulo ERROR SP_BAJA_REGALO: El regalo no existe. No se puede dar de baja. ERROR SP_BAJA_REGALO: El regalo no puede darse de baja. No es compatible su estado con la baja. ERROR SP_BAJA_REGALO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- No se puede dar de baja un regalo que no está en estado “AFEGIT_A_GRUP”.</li> <li>- Un regalo dado de baja pasa a estado “ELIMINAT”.</li> </ul>

<b>SP_ALTA_PARTICIPANTE</b>
<b>Función</b>
- Da de alta un participante en un grupo
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_id_miembro: código único del miembro de la red.</li> <li>- p_grupo: código del grup.</li> <li>- p_dedicatoria_personal: dedicatoria particular del participante.</li> </ul>
<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el miembro forma parte de la red social.</li> <li>- Comprueba que el grupo esta dado de alta.</li> <li>- Comprueba que el miembro no está repetido en el grupo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: “OK_ALTA_PARTICIPANTE”</li> <li>- Ejecución fracasada: ERROR SP_ALTA_PARTICIPANTE: El campo miembro no puede ser nulo. ERROR SP_ALTA_PARTICIPANTE: El campo grupo no puede ser nulo. ERROR SP_ALTA_PARTICIPANTE: El miembro no existe. ERROR SP_ALTA_PARTICIPANTE: El grupo no existe. ERROR SP_ALTA_PARTICIPANTE: El grupo no puede tener miembros repetidos. ERROR SP_ALTA_PARTICIPANTE: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- El identificador de un participante dentro del grupo es su e-mail.

<b>SP_BAJA_PARTICIPANTE</b>
<b>Función</b>
- Da de baja un participante.
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_id_miembro: código del miembro de la red</li> <li>- p_grupo: código del grupo</li> </ul>
<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el miembro es participante del grupo.</li> <li>- Comprueba que el miembro no tiene aportaciones a los regalos del grupo.</li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

Salida
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_BAJA_PARTICIPANTE"</li> <li>- Ejecución fracasada: ERROR SP_BAJA_PARTICIPANTE: El campo miembro no puede ser nulo. ERROR SP_BAJA_PARTICIPANTE: El campo grupo no puede ser nulo. ERROR SP_BAJA_PARTICIPANTE: El miembro no está asociado al grupo. ERROR SP_BAJA_PARTICIPANTE: El miembro no puede darse de baja. Anule las aportaciones. ERROR SP_BAJA_PARTICIPANTE: {Tipo de error} en cualquier otro caso.</li> </ul>
Consideraciones
<ul style="list-style-type: none"> <li>- No puede darse de baja un participante que tiene aportaciones a un regalo. Estas deben de anularse con su correspondiente procedimiento.</li> </ul>

<b>SP_ALTA_HOMENAJEADO</b>	
Función	- Da de alta un homenajeado en un grupo.
Parámetros de entrada	<ul style="list-style-type: none"> <li>- p_homenajeado: código del miembro de la red</li> <li>- p_grupo: código del grupo</li> </ul>
Parámetros de salida	- RSP VARCHAR2
Comprobaciones	<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el homenajeado tenga deseos.</li> <li>- Comprueba que el grupo este dado de alta.</li> <li>- Comprueba que el grupo no tenga asignado homenajeado.</li> </ul>
Salida	<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_ALTA_HOMENAJEADO"</li> <li>- Ejecución fracasada: ERROR SP_ALTA_HOMENAJEADO: El campo homenajeado no puede ser nulo. ERROR SP_ALTA_HOMENAJEADO: El campo grupo no puede ser nulo. ERROR SP_ALTA_HOMENAJEADO: El homenajeado no tiene deseos y no se puede dar de alta. ERROR SP_ALTA_HOMENAJEADO: El grupo no existe. Debe de darse de alta previamente. ERROR SP_ALTA_HOMENAJEADO: El grupo ya tiene homenajeado. Para cambiarlo debe dar de baja al actual. ERROR SP_ALTA_HOMENAJEADO: {Tipo de error} en cualquier otro caso.</li> </ul>
Consideraciones	<ul style="list-style-type: none"> <li>- Si no se indica el receptor, pero si el homenajeado, el receptor es por defecto el homenajeado.</li> </ul>

<b>SP_BAJA_HOMENAJEADO</b>	
Función	- Da de baja un homenajeado.
Parámetros de entrada	<ul style="list-style-type: none"> <li>- p_homenajeado: código del miembro de la red.</li> <li>- p_grupo: código del grupo.</li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.
- Comprueba la correlación entre grupo y homenajeadado
- Comprueba que el homenajeadado no tiene regalos en procedimiento de deseo
<b>Salida</b>
- Ejecución finalizada con éxito: "OK_BAJA_HOMENAJEADO"
- Ejecución fracasada: ERROR SP_ALTA_HOMENAJEADO: El campo homenajeadado no puede ser nulo. ERROR SP_ALTA_HOMENAJEADO: El campo grupo no puede ser nulo. ERROR SP_ALTA_HOMENAJEADO: El homenajeadado no está asignado al grupo. ERROR SP_ALTA_HOMENAJEADO: El homenajeadado tiene deseos que se están procesando como regalos. ERROR SP_BAJA_HOMENAJEADO: {Tipo de error} en cualquier otro caso.

<b>SP_ALTA_DESEO</b>
<b>Función</b>
- Da de alta un deseo.
<b>Parámetros de entrada</b>
- p_homenajeadado: código de un miembro de la red social.
- p_artículo: código de un artículo.
- p_fecha_alta: fecha en que se produce el alta del deseo.
<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
- Comprueba que ninguno de los parámetros de entrada es nulo.
- Comprueba que el miembro forma parte de la red social.
- Comprueba que el artículo está disponible.
- Comprueba que el deseo de un miembro no se repite.
<b>Salida</b>
- Ejecución finalizada con éxito: "OK_ALTA_DESEO"
- Ejecución fracasada: ERROR SP_ALTA_DESEO: El campo homenajeadado no puede ser nulo ERROR SP_ALTA_DESEO: El campo artículo no puede ser nulo. ERROR SP_ALTA_DESEO: El campo fecha de alta no puede ser nulo. ERROR SP_ALTA_DESEO: El artículo no esta disponible. ERROR SP_ALTA_DESEO: El homenajeadado no es miembro de la red social. ERROR SP_ALTA_DESEO: El deseo está repetido para ese miembro. ERROR SP_ALTA_DESEO: {Tipo de error} en cualquier otro caso.

<b>SP_BAJA_DESEO</b>
<b>Función</b>
- Da de baja un deseo.
<b>Parámetros de entrada</b>
- p_id_deseo: código del deseo.
<b>Parámetros de salida</b>
- RSP VARCHAR2



**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros de entrada es nulo.</li> <li>- Comprueba que el deseo existe.</li> <li>- Comprueba que el deseo no se esté procesando como regalo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_BAJA_DESEO"</li> <li>- Ejecución fracasada: ERROR SP_BAJA_DESEO: El campo deseo no puede ser nulo. ERROR SP_BAJA_DESEO: El deseo no existe. No se puede dar de baja. ERROR SP_BAJA_DESEO: El deseo se está procesando como regalo. No se puede dar de baja. ERROR SP_BAJA_DESEO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- No se puede dar de baja un deseo que ha comenzado a tramitarse como regalo.</li> </ul>

<b>SP_MODIFICACIÓN_DESEO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Modifica el deseo solicitado de un miembro de la red.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_id_deseo: código del deseo.</li> <li>- p_nuevo_artículo: código del nuevo artículo del deseo.</li> </ul>
<b>Parámetros de salida</b>
<ul style="list-style-type: none"> <li>- RSP VARCHAR2</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros de entrada es nulo.</li> <li>- Comprueba que el deseo existe.</li> <li>- Comprueba que el deseo no se esté procesando como regalo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_MODIFICACIÓN_DESEO"</li> <li>- Ejecución fracasada: ERROR SP_MODIFICACIÓN_DESEO: El campo deseo no puede ser nulo. ERROR SP_MODIFICACIÓN_DESEO: El campo nuevo_artículo no puede ser nulo. ERROR SP_MODIFICACIÓN_DESEO: El deseo no existe. No se puede modificar. ERROR SP_MODIFICACIÓN_DESEO: El nuevo artículo no existe. No se puede modificar. ERROR SP_MODIFICACIÓN_DESEO: El deseo se está procesando como regalo. No se puede modificar. ERROR SP_MODIFICACIÓN_DESEO: {Tipo de error} en cualquier otro caso.</li> </ul>

<b>SP_ALTA_TRANSACCIÓN</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Da de alta una transacción en la tabla transacciones.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_regalo: código de un regalo.</li> <li>- p_aportador: código de un miembro de la red social que realiza una aportación.</li> <li>- p_cantidad: cantidad aportada al regalo por el aportador.</li> <li>- p_forma_pago: forma de pago elegida por el aportado.</li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

- p_visible: indica si la aportación será visible en el grupo.
<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que el miembro es miembro de la red social, y es participante del grupo</li> <li>- Comprueba que el regalo existe.</li> <li>- Comprueba que el regalo esta en un estado compatible con la aportación.</li> <li>- Comprueba que la cantidad aportada es superior a 0.</li> <li>- Comprueba que la cantidad aportada no es superior a la necesaria.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: "OK_ALTA_TRANSACCIÓN"</li> <li>- Ejecución fracasada: ERROR SP_ALTA_TRANSACCIÓN: El campo regalo no puede ser nulo. ERROR SP_ALTA_TRANSACCIÓN: El campo aportador no puede ser nulo. ERROR SP_ALTA_TRANSACCIÓN: El campo cantidad no puede ser nulo. ERROR SP_ALTA_TRANSACCIÓN: El campo forma_pago no puede ser nulo. ERROR SP_ALTA_TRANSACCIÓN: El campo visible no puede ser nulo. ERROR SP_ALTA_TRANSACCIÓN: El regalo no está disponible. Compruebe que ha sido añadido como regalo. ERROR SP_ALTA_TRANSACCIÓN: El regalo no permite aportaciones. Su estado es incompatible con una aportación. ERROR SP_ALTA_TRANSACCIÓN: El aportador no es miembro de la red social. ERROR SP_ALTA_TRANSACCIÓN: El aportador no pertenece a ningún grupo. ERROR SP_ALTA_TRANSACCIÓN: El aportador no pertenece al grupo del que selecciono el regalo. ERROR SP_ALTA_TRANSACCIÓN: La cantidad aportada debe ser superior a 0. ERROR SP_ALTA_TRANSACCIÓN: La cantidad aportada es excesiva. Sólo son necesarios <i>n</i> euros. ERROR SP_ALTA_TRANSACCIÓN: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- Sólo se permiten aportaciones a regalos que están en estado 'AFEGIT_A_UN_GRUP'.

<b>SP_BAJA_TRANSACCIÓN</b>
<b>Función</b>
- Da de baja una transacción en la tabla transacciones.
<b>Parámetros de entrada</b>
- p_id_transacción: código de la transacción que se quiere dar de baja.
<b>Parámetros de salida</b>
- RSP VARCHAR2
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que ninguno de los parámetros obligatorios de entrada es nulo.</li> <li>- Comprueba que existe la transacción a dar de baja.</li> <li>- Comprueba que no se da de baja la transacción más de una vez.</li> <li>- Comprueba que el regalo está en un estado adecuado.</li> </ul>
<b>Salida</b>
- Ejecución finalizada con éxito:

<p>“OK_BAJA_TRANSACCIÓN”</p> <ul style="list-style-type: none"> <li>- Ejecución fracasada:  ERROR SP_BAJA_TRANSACCIÓN: El campo transacción no puede ser nulo.  ERROR SP_BAJA_TRANSACCIÓN: No existe ninguna transacción con esa id.  ERROR SP_BAJA_TRANSACCIÓN: No se puede dar de baja una transacción más de una vez.  ERROR SP_BAJA_TRANSACCIÓN: El estado del regalo es incompatible con la anulación de la transacción.  ERROR SP_BAJA_TRANSACCIÓN: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Sólo se permiten baja de transacción si el regalo esta en estado ‘AFEGIT_A_UN_GRUP’.</li> </ul>

### 4.3 PROCEDIMIENTOS DE CONSULTA

Cada consulta planteada en el segundo punto del apartado R5 del enunciado dispone de una función almacenada. En este apartado se incluyen las especificaciones de las mentadas funciones.

<b>CONSULTA_A</b>
<b>Función</b>
- Dado un miembro de la red social: el listado de sus regalos deseados; Incluyendo el precio y el proveedor asociado, así como la fecha de alta del deseo.
<b>Parámetros de entrada</b>
- p_id_miembro NUMBER: código de un miembro de la red social.
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> <li>- Comprueba que el miembro tiene deseos.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Descripción del artículo.</li> <li>• Precio del artículo.</li> <li>• Proveedor del artículo.</li> <li>• Fecha alta deseo.</li> </ul> </li> <li>- Ejecución fracasada:  ERROR CONSULTA_A: El campo <i>id_miembro</i> no puede ser nulo.  ERROR CONSULTA_A: No existe ningún homenajeado con deseos con esa id.  ERROR CONSULTA_A: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- En caso de no encontrar respuesta devuelve una consulta vacía.

<b>CONSULTA_B</b>
<b>Función</b>
- Mostrar el listado de todos los grupos de participantes que todavía tienen algún regalo pendiente, es decir, que todavía no han recogido los fondos necesarios para que sus regalos se hayan podido adquirir. Indicando también la suma total de todos los regalos, el importe que todavía falta por conseguir y el porcentaje respecto al importe total que se ha conseguido.
<b>Parámetros de entrada</b>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

- Ningún parámetro.
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Código id del grupo.</li> <li>• Importe total.</li> <li>• Importe restante.</li> <li>• Porcentaje respecto al importe total que se ha conseguido.</li> </ul> </li> <li>- Ejecución fracasada: ERROR CONSULTA_B: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- En caso de no encontrar respuesta devuelve una consulta vacía.

<b>CONSULTA_C</b>
<b>Función</b>
- Dada una categoría: el listado de todos los regalos asociados, incluyendo los proveedores que los proporcionan.
<b>Parámetros de entrada</b>
- p_id_categoría NUMBER: código de una categoría de artículos.
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> <li>- Comprueba que la categoría tiene regalos asociados.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Descripción del regalo.</li> <li>• Nombre del proveedor.</li> </ul> </li> <li>- Ejecución fracasada: ERROR CONSULTA_C: El campo <i>id_categoría</i> no puede ser nulo. ERROR CONSULTA_C: No existe ningún regalo para esa categoría. ERROR CONSULTA_C: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- En caso de no encontrar respuesta devuelve una consulta vacía.

<b>CONSULTA_D</b>
<b>Función</b>
- Dado un usuario de la red social: el listado detallado de todos los regalos que ha recibido.
<b>Parámetros de entrada</b>
- p_id_miembro NUMBER: código de un miembro de la red social.
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> <li>- Comprueba que el miembro tiene deseos.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Código id del artículo.</li> <li>• Descripción del artículo.</li> <li>• Precio del artículo.</li> <li>• Categoría del artículo.</li> <li>• Proveedor del artículo.</li> </ul> </li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>- Ejecución fracasada: ERROR CONSULTA_D: El campo <i>id_miembro</i> no puede ser nulo. ERROR CONSULTA_D: No existe ningún homenajeado con deseos con esa id. ERROR CONSULTA_D: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- En caso de no encontrar respuesta devuelve una consulta vacía.

<b>CONSULTA_E</b>
<b>Función</b>
- Mostrar el listado de las personas que se encuentran por encima de la media a nivel de las aportaciones económicas que han destinado a regalos grupales.
<b>Parámetros de entrada</b>
- Ningún parámetro.
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Nombre del miembro.</li> <li>• Primer apellido del miembro.</li> <li>• Segundo apellido del miembro.</li> </ul> </li> <li>- Ejecución fracasada: ERROR CONSULTA_E: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- La media de aportaciones no se acota a ningún periodo concreto.</li> <li>- En caso de no encontrar respuesta devuelve una consulta vacía.</li> </ul>

<b>CONSULTA_F</b>
<b>Función</b>
- Dado un año: el listado de las 10 ciudades donde más regalos han recibido las persona homenajeadas durante el años indicado, así como el valor económico total correspondiente a la suma de los regalos en cada una de estas ciudades
<b>Parámetros de entrada</b>
- p_año INTEGER: año para el que se hace la consulta.
<b>Comprobaciones</b>
- Comprueba que el parámetro de entrada no sea nulo.
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Importe total de todos los regalos recibidos en una ciudad.</li> <li>• Nombre de la ciudad.</li> </ul> </li> <li>- Ejecución fracasada: ERROR CONSULTA_F: El campo <i>año</i> no puede ser nulo. ERROR CONSULTA_F: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- En caso de no encontrar respuesta devuelve una consulta vacía.

#### 4.4 PROCEDIMIENTOS DEL MÓDULO ESTADÍSTICO

El módulo estadístico atiende a los requisitos expuestos en el punto R7 del enunciado del TFC. Este punto especifica una serie de estadísticas que deben actualizarse de forma constante en función del uso que se realice de la base de

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

datos. Funcionan, por tanto, de forma sincronizada con los procedimientos de alta, baja y modificación.

Este modulo consta de 9 tablas y una cantidad igual de procedimientos. Los procedimientos se encargan de pre calcular el dato estadístico y discriminar si es relevante o no su conservación en las tablas estadísticas. Por último, una consulta sobre la tabla debe permitir visualizar el dato que se considere pertinente.

Las especificaciones de las tablas se encuentran en el anexo de este documento y a continuación se detallan cuales son los procedimientos que componen este bloque.

<b>SP_EST1_VECES_REGALADO</b>	
<b>Función</b>	<ul style="list-style-type: none"> <li>- Este procedimiento calcula el número de veces que se ha regalado un artículo determinado a diferentes miembros de la red.</li> <li>- El número de veces se vincula a un producto y un años concreto.</li> <li>- Añade o actualiza información en la tabla EST1_VECES_REGALADO.</li> </ul>
<b>Parámetros de entrada</b>	<ul style="list-style-type: none"> <li>- p_producto NUMBER: código de un producto.</li> <li>- p_año NUMBER: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>	<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> <li>- Comprueba que existe el producto.</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Código de producto.</li> <li>• Año.</li> <li>• Número de veces que se ha regalado el artículo.</li> </ul> </li> <li>- Ejecución fracasada: <p>ERROR SP_EST1_VECES_REGALADO : El campo <i>p_producto</i> no puede ser nulo.</p> <p>ERROR SP_EST1_VECES_REGALADO : El campo <i>p_año</i> no puede ser nulo.</p> <p>ERROR SP_EST1_VECES_REGALADO : No existe ningún producto con ese código.</p> <p>ERROR SP_EST1_VECES_REGALADO: {Tipo de error} en cualquier otro caso.</p> </li> </ul>
<b>Consideraciones</b>	<ul style="list-style-type: none"> <li>- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.</li> </ul>

<b>SP_EST2_TOTAL_REGALOS</b>	
<b>Función</b>	<ul style="list-style-type: none"> <li>- Este procedimiento calcula el número de regalos totales que se han adquirido utilizando la aplicación.</li> <li>- El número de regalos se vincula a un año concreto.</li> <li>- Añade o actualiza información en la tabla SP_EST2_TOTAL_REGALOS.</li> </ul>
<b>Parámetros de entrada</b>	<ul style="list-style-type: none"> <li>- p_año NUMBER: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>	<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> </ul>
<b>Salida</b>	<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito:</li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>• Año.</li> <li>• Número total de regalos adquiridos.</li> </ul> <p>- Ejecución fracasada: ERROR SP_EST2_TOTAL_REGALOS: El campo <i>p_año</i> no puede ser nulo. ERROR SP_EST2_TOTAL_REGALOS {Tipo de error} en cualquier otro caso.</p>
<b>Consideraciones</b>
- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.

<b>SP_EST3_IMPORTE_TOTAL</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento calcula el importe total en regalos que se han adquirido utilizando la aplicación</li> <li>- El importe total se vincula a un año concreto.</li> <li>- Añade o actualiza información en la tabla SP_EST3_IMPORTE_TOTAL.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- <i>p_producto</i> NUMBER: código de un producto.</li> <li>- <i>p_año</i> NUMBER: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que los parámetros de entrada no sean nulos.</li> <li>- Comprueba que existe el producto.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Importe total de los regalos adquiridos.</li> </ul> </li> <li>- Ejecución fracasada: ERROR SP_EST3_IMPORTE_TOTAL : El campo <i>p_producto</i> no puede ser nulo. ERROR SP_EST3_IMPORTE_TOTAL: El campo <i>p_año</i> no puede ser nulo. ERROR SP_EST3_IMPORTE_TOTAL: No existe ningún producto con ese código. ERROR SP_EST3_IMPORTE_TOTAL: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.

<b>SP_EST4_PORCENTAJE_COMPLETADO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento calcula el porcentaje de regalos pagados respecto a los que no consiguen reunir los fondos necesarios para su adquisición.</li> <li>- El porcentaje se vincula a un año concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST4_PORCENTAJE_COMPLETADO.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- <i>p_año</i> NUMBER: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> </ul> </li> </ul>



**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>• Porcentaje de regalos pagados respecto a los que no completan la recaudación.</li> </ul>
<ul style="list-style-type: none"> <li>- Ejecución fracasada: ERROR SP_EST4_PORCENTAJE_COMPLETADO: El campo <i>p_año</i> no puede ser nulo. ERROR SP_EST4_PORCENTAJE_COMPLETADO: {Tipo de error} en cualquier otro caso. ERROR SP_EST4_PORCENTAJE_COMPLETADO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.</li> </ul>

<b>SP_EST5_MEDIA_GASTO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento calcula la media de gasto de cada miembro de la red social.</li> <li>- La media se vincula a un año concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST5_MEDIA_GASTO .</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- <i>p_producto</i> NUMBER: código de un producto.</li> <li>- <i>p_año</i> NUMBER: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> <li>- Comprueba que existe el producto.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Media de gasto en regalos de los miembros de la red</li> </ul> </li> <li>- Ejecución fracasada: ERROR SP_EST5_MEDIA_GASTO: El campo <i>p_producto</i> no puede ser nulo. ERROR SP_EST5_MEDIA_GASTO: El campo <i>p_año</i> no puede ser nulo. ERROR SP_EST5_MEDIA_GASTO: No existe ningún producto con ese código. ERROR SP_EST5_MEDIA_GASTO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.</li> <li>- Dado que no hay fechas que daten el alta y la baja de socios no se puede establecer cuantos socios tiene la red en un año concreto, este factor resta consistencia a la estadística.</li> </ul>

<b>SP_EST6_MAS_HOMENAJEADO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento obtiene el identificador del miembro que ha recibido más regalos</li> <li>- El miembro se vincula a un año y un mes concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST6_MAS_HOMENAJEADO .</li> </ul>
<b>Parámetros de entrada</b>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>- p_homenajead: NUMBER: código de un miembro homenajead con un regalo.</li> <li>p_fecha DATE: año en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que los parámetros de entrada no sean nulos.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Mes.</li> <li>• Código del miembro más homenajead.</li> <li>• Número de veces homenajead.</li> </ul> </li> <li>- Ejecución fracasada: <p>ERROR SP_EST6_MAS_HOMENAJEADO: El campo <i>homenajead</i> no puede ser nulo.</p> <p>ERROR SP_EST6_MAS_HOMENAJEADO: El campo <i>fecha</i> no puede ser nulo.</p> <p>ERROR SP_EST6_MAS_HOMENAJEADO: {Tipo de error} en cualquier otro caso.</p> </li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.</li> </ul>

<b>SP_EST7_MAS_GASTADOR</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento obtiene el identificador del miembro que más gasto en regalos a realizado</li> <li>- El miembro se vincula a un año y un mes concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST7_MAS_GASTADOR.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_fecha: : fecha en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Mes.</li> <li>• Código del miembro más gastador.</li> <li>• Cantidad aportada por el miembro más gastador.</li> </ul> </li> <li>- Ejecución fracasada: <p>ERROR SP_EST7_MAS_GASTADOR: El campo <i>fecha</i> no puede ser nulo.</p> <p>ERROR SP_EST7_MAS_GASTADOR: {Tipo de error} en cualquier otro caso.</p> </li> </ul>
<b>Consideraciones</b>
<ul style="list-style-type: none"> <li>- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.</li> </ul>

<b>SP_EST8_MAS_PARTICIPATIVO</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento obtiene el identificador del miembro que ha participado en más regalos diferentes.</li> </ul>

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

<ul style="list-style-type: none"> <li>- El miembro se vincula a un año y un mes concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST8_MAS_PARTICIPATIVO.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_fecha: : fecha en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Mes.</li> <li>• Código del miembro más participativo.</li> <li>• Número de veces que ha participado en un regalo</li> </ul> </li> <li>- Ejecución fracasada: ERROR SP_EST8_MAS_PARTICIPATIVO: El campo <i>fecha</i> no puede ser nulo. ERROR SP_EST8_MAS_PARTICIPATIVO: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.

<b>SP_EST9_GRUPO_MAS_GASTADOR</b>
<b>Función</b>
<ul style="list-style-type: none"> <li>- Este procedimiento obtiene el identificador del grupo que ha gastado más en regalos.</li> <li>- El grupo se vincula a un año y un mes concreto.</li> <li>- Añade o actualiza información contenida en la tabla SP_EST9_GRUPO_MAS_GASTADOR.</li> </ul>
<b>Parámetros de entrada</b>
<ul style="list-style-type: none"> <li>- p_fecha: : fecha en que se produce la adquisición del producto.</li> </ul>
<b>Comprobaciones</b>
<ul style="list-style-type: none"> <li>- Comprueba que el parámetro de entrada no sea nulo.</li> </ul>
<b>Salida</b>
<ul style="list-style-type: none"> <li>- Ejecución finalizada con éxito: <ul style="list-style-type: none"> <li>• Año.</li> <li>• Mes.</li> <li>• Código del grupo.</li> <li>• Cantidad aportada por el grupo</li> </ul> </li> <li>- Ejecución fracasada: ERROR SP_EST9_GRUPO_MAS_GASTADOR: El campo <i>fecha</i> no puede ser nulo. ERROR SP_EST9_GRUPO_MAS_GASTADOR: {Tipo de error} en cualquier otro caso.</li> </ul>
<b>Consideraciones</b>
- Este procedimiento se ejecuta de forma automática cuando se ha producido una transacción que da lugar a la modificación de un regalo, de forma tal que este último pasa de estado 'AFEGIT_UN_GRUP' a 'PAGAMENT_COMPLETAT'.

## 5. PLAN DE CONTINGENCIAS

En este capítulo se referencian las actuaciones que forman el plan de contingencia, requisito explícito del personal docente.

Se proponen, por tanto, una serie de procedimientos destinados a subsanar y corregir las incidencias que puedan surgir durante la elaboración del presente trabajo.

De forma básica son tres las incidencias que pueden entorpecer el desarrollo de la asignatura:

- Avería del punto de trabajo.
- Enfermedad.
- Indisposición horaria.

En esta breve introducción reseñaremos los aspectos relativos al plan de contingencia que atañen al último punto, por ser este el que permite aplicar una mayor cuota de previsión. Los otros puntos también tendrán mención, pero no permiten definir ninguna actuación preventiva.

La variedad de soportes y métodos de almacenamiento deben asegurar de forma robusta la eliminación de cualquier incidencia que ocasione pérdida de información. En otro orden de cosas, la utilización de máquinas virtuales asegura una rápida recomposición del entorno de trabajo.

Es, por tanto, una política de buenas prácticas realizar de forma rutinaria copias de seguridad de la información relativa al desarrollo del proyecto y disponer de copias del entorno de la máquina virtual.

La realización del trabajo utilizará para la salvaguarda de datos la aplicación *Time Machine* que viene instalada con el sistema operativo *Mac OS X* y asegura en sí misma un plan de contingencia frente los inconveniente por avería del puesto de trabajo.

El mismo programa genera las copias de seguridad integra de todo el sistema de la siguiente forma:

- Copia de seguridad cada hora durante las últimas 24 horas
- Copias de seguridad diarias del último mes.
- Copias de seguridad semanales de los meses anteriores.

Ante las indisposiciones generadas por enfermedad o indisposición horaria sólo en caso de fuerza mayor se contempla un reajuste de la planificación. Estas modificaciones se deben comunicar y consensuar con el personal docente que debe dar su aprobación si lo cree oportuno.

A continuación se detallan de forma más específica los riesgos en la siguiente tabla:

DESCRIPCIÓN DEL RIESGO	PROBABILIDAD	IMPACTO	CONTRAMEDIDA PREVENTIVA-CORRECTIVA
Indisponibilidad por enfermedad leve.	BAJA	MODERADO/NULO	Corregir en el transcurso de los días posteriores el déficit de progreso generado.
Indisponibilidad por enfermedad grave.	BAJA	MUY ALTO	Consultar con el personal docente si hay alguna posibilidad viable de corregir la planificación.
Indisponibilidad esporádica por motivos laborales.	ALTA	MODERADO/NULO	Corregir en el transcurso de los días posteriores el déficit de progreso generado.
Indisponibilidad reiterada por motivos laborales.	ALTA	ALTO	Consultar con el personal docente si hay alguna posibilidad viable de corregir la planificación.
Indisponibilidad del puesto de trabajo por fallo de hardware.	BAJA	MODERADO	Recuperar copia de seguridad, importar datos a nuevo equipo.
Indisponibilidad del puesto de trabajo por fallo de software.	BAJA	MODERADO	Recuperar copia de seguridad y reinstalar el software necesario.
Imprevistos derivados de la falta de conexión a internet en fechas críticas: entregas, tutorías...	BAJA	MODERADO	Disponer de conexión 3G como alternativa que permita una conectividad auxiliar.
Imprevistos derivados el corte de suministro eléctrico en fechas críticas: entregas, tutorías...	BAJA	MODERADO	Disponer de SAI que no interrumpa el funcionamiento del equipo. Si los equipos son portátiles se pueden trasladar a un punto de alimentación no afectado.

## 6. PLAN DE PRUEBAS

El producto entregado integra un plan de pruebas destinado a comprobar el correcto funcionamiento del producto implementado.

Las pruebas se han dividido de igual forma que los bloques a implementar. De esta forma se dispone de una parte destinada a los procesos de alta, baja y modificación, uno para las consultas y, por último, uno para el apartado estadístico.

## 6.1 CARGA INICIAL DE DATOS

Para poder alimentar las pruebas de los procedimientos implementados ha sido necesario realizar una carga inicial de datos sobre las tablas auxiliares.

Las tablas afectada por el bloque de carga inicial de datos son: Categoría, Proveedor, Artículo, Miembro y Método.

Las especificaciones de estas tablas están disponibles el apartado 3 Diseño y 10 Anexo.

Después de esta carga inicial el producto dispone de una cantidad suficiente de datos y queda en disposición de poder utilizarlos en los procesos que se implementaron posteriormente.

La carga inicial se encuentra disponible dentro de la carpeta *Creación de tabla-Carga inicial* con el nombre de *Carga\_Inicial.sql*.

## 6.2 PRUEBAS DE ALTA, BAJA Y MODIFICACIÓN

Una vez se dispone de un conjunto de datos inicial suficientemente amplio y variado, es posible realizar las pruebas de los procedimientos de alta, baja y modificación.

Según requerimiento del enunciado del TFC una prueba acabada con éxito debe devolver un mensaje de 'OK' y una prueba fracasada debe devolver una mensaje de error más el tipo de error.

El bloque de pruebas se encuentra disponible dentro de la carpeta *Pruebas* con el nombre *Pruebas\_ABM.sql*.

Cada procedimiento dispone de sus propias pruebas que pueden ser coincidentes con las de otros procesos, o que por su función específica han requerido pruebas destinadas a la comprobación de aspectos singulares aplicables a su funcionamiento.

De forma general todos los procesos comprueban que:

- El dato introducido no sea nulo.
- El dato introducido existe previamente si procede de otra tabla.
- El dato no se introduce de forma repetida o duplicada.
- El dato se puede modificar o borrar si presenta una agregación de condiciones que deja a la base de datos en un estado inconsistente.

No obstante son, tal como se comentaba anteriormente, muchas las ocasiones en que las funcionalidades del proceso han marcado la elaboración de pruebas destinadas a salvaguardar la coherencia de su operatividad.

A nivel de ejemplo podemos indicar:

- En el proceso Alta Transacción:
  - Se prueba si es posible aportar a un regalo una cantidad superior a la necesaria para completar el importe total del regalo.
  - Se prueba si es posible realizar una aportación a un regalo que ya ha reunido todos los fondos necesarios.

- En el proceso Baja Transacción:
  - Se prueba si es posible solicitar la devolución de un importe asociado a un regalo que está en estado pagado y, por tanto, generará de forma automática unos procesos de tramitación de envío.
  - Se prueba que el abono del importe de una transacción no se realice de forma repetida.
- En el proceso Alta Homenajeado:
  - Se comprueba que el miembro añadido como homenajeado tenga algún deseo susceptible de ser regalado.

Todas estas pruebas son fácilmente deducibles si se atiende al punto 4 de este documento dedicado a implementación y se observan los apartados dedicados a las salidas de cada proceso.

En ocasiones se ha tomado la decisión de añadir al tipo de error información que no sólo permita saber de que error se trata, si no también como operar de nuevo sin volver a incurrir en él.

Para terminar este apartado se adjunta una muestra de las salidas obtenidas en el proceso de prueba. La salida completa de este proceso se puede localizar en la carpeta *Pruebas* con el nombre *Puebas\_ABM.txt*.

```
--ALTA DE UN HOMENAJEADO QUE NO TIENE DESEOS
SP_ALTA_HOMENAJEADO ('44','4',RSP);
DBMS_OUTPUT.PUT_LINE(RSP);
```

ERROR SP\_ALTA\_HOMENAJEADO: El homenajeado no tiene deseos y no puede darse de alta

-----

```
--ALTA DE UN HOMENAJEADO EN UN GRUPO INEXISTENTE
SP_ALTA_HOMENAJEADO ('15','100',RSP);
DBMS_OUTPUT.PUT_LINE(RSP);
```

ERROR SP\_ALTA\_HOMENAJEADO: El grupo no existe. Debe de darse de alta previamente

-----

```
--ALTA TRANSACCIÓN DE UNA CANTIDAD QUE EXCEDE LO NECESARIO
SP_ALTA_TRANSACCIÓN('6','20','10','14-05-2014','2','SI',RSP);
DBMS_OUTPUT.PUT_LINE(RSP);
```

ERROR SP\_ALTA\_TRANSACCIÓN: La cantidad aportada es excesiva. Sólo son necesarios 6,5 euros

### 6.3 PRUEBAS ESTADÍSTICAS

Las pruebas del modulo estadístico mantienen algunos aspectos comunes a las realizadas en el apartado anterior, porque deben cumplir los mismos requerimientos expresados en cuanto a los mensajes de salida.

No obstante, para este bloque de pruebas el objetivo es comprobar si los resultados obtenidos en las tablas estadísticas son coherentes con las modificaciones de ABM que se efectúan en la base de datos.

A continuación se muestra un repaso de los diferentes resultados obtenidos. La salida completa de este proceso se puede localizar en la carpeta *Pruebas* con el nombre *Pruebas\_Estadísticas.txt*.

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.1

```
> SELECT *FROM EST1_VECES_REGALADO WHERE PRODUCTO=5 AND AÑO=2014
```

PRODUCTO	AÑO	NUMERO_VECES
----------	-----	--------------

-----

5	2014	1
---	------	---

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.2

```
> SELECT *FROM EST2_TOTAL_REGALOS WHERE AÑO=2014
```

AÑO	TOTAL_REGALOS
-----	---------------

-----

2014	5
------	---

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.3

```
> SELECT *FROM EST3_IMPORTE_TOTAL WHERE AÑO=2014
```

AÑO	IMPORTE_TOTAL
-----	---------------

-----

2014	32,5
------	------

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.4

```
> SELECT *FROM EST4_PORCENTAJE_COMPLETADO WHERE AÑO=2014
```

AÑO	PORCENTAJE
-----	------------

-----

2014	60
------	----

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.5



```
> SELECT *FROM EST5_MEDIA_GASTO WHERE AÑO=2014
```

```

AÑO MEDIA_GASTO
-----
2014          ,65

```

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.6

```
> SELECT *FROM EST6_MAS_HOMENAJEADO WHERE MES=7 AND AÑO=2014
```

```

AÑO      MES ID_MIEMBRO  N_VECES
-----
2014      7         1         3

```

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.7

```
> SELECT *FROM EST7_MAS_GASTADOR WHERE MES=6 AND AÑO=2014
```

```

AÑO      MES ID_MIEMBRO  CANTIDAD
-----
2014      6         29      13,75

```

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.8

```
> SELECT *FROM EST8_MAS_PARTICIPATIVO WHERE MES=6 AND AÑO=2014
```

```

AÑO      MES ID_MIEMBRO  N_VECES
-----
2014      6         15         2
2014      6         29         2

```

#### COMPROBACIÓN DE LA ESTADÍSTICA R6.9

```
> SELECT *FROM EST9_GRUPO_MAS_GASTADOR WHERE MES=6 AND AÑO=2014
```

```

AÑO      MES  ID_GRUPO  TOTAL
-----
2014      6         5      18,75

```

### **6.4 PRUEBAS CONSULTAS**

Las pruebas de las consultas están destinadas a comprobar el correcto funcionamiento de estas.

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

El bloque de pruebas se encuentra disponible en la carpeta *Prueba* con el nombre *Pruebas\_Consultas.sql*.

A continuación se muestran algunas de las comprobaciones realizadas:

**COMPROBACIÓN CONSULTA A**

```
SELECT *FROM TABLE(CONSULTAA('1'))
```

REGALO	PRECIO	PROVEEDOR	FECHA_ALTA
-----	-----	-----	-----
Libro 1	6	Libreria SA	01/01/14
Libro 2	6,25	Libreria SA	02/01/14
Libro 3	6,5	Libreria SA	03/01/14
Libro 5	7	Libreria SA	01/04/14

**COMPROBACIÓN CONSULTA B**

```
SELECT *FROM TABLE(CONSULTAB());
```

GRUPO	IMPORTE_TOTAL	IMPORTE_RESTANTE	PORCENTAJE
-----	-----	-----	-----
4	16,5	14	15,15
5	7	7	0
6	7,25	7,25	0
7	7	7	0

**COMPROBACIÓN CONSULTA C**

```
SELECT *FROM TABLE(CONSULTABC('1'));
```

REGALO	PROVEEDOR
-----	-----
Libro 3	Libreria SA
Libro 1	Libreria SA
Libro 4	Libreria SA
Libro 2	Libreria SA
Libro 5	Libreria SA
Libro 6	Libreria SA
Libro 7	Libreria SA

### COMPROBACIÓN CONSULTA D

```
SELECT *FROM TABLE(CONSULTA('1'));
```

ARTÍCULO	DESCRIPCIÓN	PRECIO	CATEGORÍA	PROVEEDOR
1	Libro 1	6	Libro	Librería SA
2	Libro 2	6,25	Libro	Librería SA
3	Libro 3	6,5	Libro	Librería SA

### COMPROBACION CONSULTA E

```
SELECT *FROM TABLE(CONSULTAE());
```

NOMBRE	APELLIDO1	APELLIDO2	IMPORTE
Rosario	Coconut	Trigo	13,75
Mario	Rodríguez	Raspall	9,5
Javier	Pi	Negre	3,5
Ricardo	Cardó	Lois	3
Isidro	Diéguez	Rico	2,5
Simeon	Toro	Bravo	2

### COMPROBACIÓN CONSULTA F

```
SELECT *FROM TABLE(CONSULTAF('2014'));
```

TOTAL CIUDAD
32,5 ciudad 1

## 6.4 COMPROBACIÓN LOGS

Realizaremos, por último una mención a la tabla Logs que, aunque no es propiamente una prueba, merece especial atención porque recoge toda la información producida por las actividades de altas, bajas y modificaciones, así como la derivada del uso del bloque estadístico y las consultas. Esta tabla se puede consultar en cualquier momento.

## 7. RECURSOS NECESARIOS Y VALORACIÓN ECONÓMICA

Este apartado está destinado a ofrecer un presupuesto que determine el coste del proyecto, también ofrece una relación de los recursos necesarios para su desarrollo.

### 7.1 RECURSOS NECESARIOS

#### 7.1.1 RECURSOS HUMANOS

Los recursos humanos hacen referencia al conjunto de trabajadores y los diferentes roles que desempeñan estos en la elaboración del proyecto.

El número de perfiles que intervienen en un proyecto pueden ser muy variado, pero de forma específica reseñaremos la intervención de dos tipos de profesionales:

- Analista. Se encarga de las tareas de análisis, diseño y documentación del producto.
- Programador. Su labor es poner en marcha el puesto de trabajo, realizar la implementación del producto y hacer la pruebas correspondientes que validen la calidad del producto.

Existen perfiles específicos que también podrían aplicarse en la valoración, como por ejemplo la utilización de un profesional especialista en generar la documentación asociada al proyecto o un probador de producto.

No obstante con los perfiles reseñados al comienzo del apartado se puede realizar la estimación del coste del proyecto.

#### 7.1.2 RECURSOS *HARDWARE* Y *SOFTWARE*

Esta parte recoge los aspectos más comunes relacionados con los recursos de *hardware* y *software* necesarios para la realización del proyecto.

- *Hardware*. Para nuestro caso disponemos de un equipo con la siguiente configuración:

- Procesador 3.2 GHz Intel Core i3
- Memoria Ram 4G.
- Disco duro 1 TB

- *Software*. Para nuestro caso utilizaremos los siguientes programas:

- Sistema operativo Mac OSX versión 10.6.8
- Microsoft Word 2011
- Microsoft PowerPoint 2011
- VMware Fusión
- Sistema operativo Windows Xp

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

- Oracle 11g EX
- SQL Developer

## 7.2 VALORACIÓN ECONÓMICA

En función de la planificación podemos determinar el numero de horas empleado, si como media se dedican a la elaboración del TFC dos horas por jornada.

Nombre	Fecha de ini...	Fecha de fin	Duración
☐ • Trabajo fin de carrera	26/02/14	15/06/14	110
☐ • Plan de trabajo	26/02/14	16/03/14	19
☐ • Análisis y diseño	17/03/14	14/04/14	29
• Revisión de requisitos	17/03/14	22/03/14	6
• Diseño conceptual	23/03/14	28/03/14	6
• Diseño lógico	29/03/14	3/04/14	6
• Diseño físico	4/04/14	9/04/14	6
• Redacción, revisión y entrega PAC2	10/04/14	14/04/14	5
• PAC2	14/04/14	14/04/14	0
☐ • Implementación y pruebas	14/04/14	11/05/14	28
• Creación de tablas	14/04/14	18/04/14	5
• Creación de procedimientos y disparadores	19/04/14	23/04/14	5
• Elaboración del modulo estadístico	24/04/14	28/04/14	5
• Elaboración de pruebas	29/04/14	3/05/14	5
• Ajustes sobre los resultados de pruebas	4/05/14	8/05/14	5
• Redacción, revisión y entrega PAC3	9/05/14	11/05/14	3
• PAC3	11/05/14	11/05/14	0
☐ • Entrega final	12/05/14	15/06/14	35
• Elaboración de memoria	12/05/14	15/06/14	35
• Elaboración de presentación	12/05/14	15/06/14	35
• ENTREGA FINAL	15/06/14	15/06/14	0

### Coste del TFC

Mano de obra	Concepto	Precio hora	Jornadas	horas	total
Analista	Análisis	30 €	6	12	360 €
Analista	Diseño	30 €	18	36	1080 €
Programador	Documentación	25 €	35	70	2100 €
Programador	Implementación	25 €	15	30	750 €
Programador	Pruebas	25 €	5	10	250 €
Programador	Ajustes	25 €	5	10	250 €
				Importe	4790€
				IVA 21%	1005€
				Total	5795€

Se eliminan de la elaboración del coste productos intermedios como la realización de las PACS que sólo atañen al aspecto académico del proyecto.

En otro orden de cosas, aspectos como costes en software, básicamente licencias, o hardware para la elaboración del producto corren a cargo de la empresa contratada.

## 8. CONCLUSIONES

### 8.1. CONCLUSIONES GENERALES

Tras la realización del trabajo se pueden obtener las siguientes conclusiones:

Se han logrado alcanzar los objetivos propuestos en el enunciado del trabajo. El alumno ha conseguido dar solución a la mayoría de cuestiones propuestas y también ha adquirido conocimientos adicionales usados de forma específica para la realización del presente trabajo.

El producto final implementado presenta fiabilidad frente al plan de pruebas, por tanto podemos deducir que tiene un grado de calidad aceptable.

Quedan patentes la importancia de las fases de planificación y diseño para poder alcanzar los objetivos marcados. Estas fases han presentado en ocasiones demasiadas modificaciones que han obstaculizado el correcto discurrir de la elaboración del TFC.

El seguimiento del método de desarrollo llamado *ciclo de vida clásica o en cascada* ha evidenciado que cualquier deficiencia en etapas de análisis o diseño acababa repercutiendo de forma severa en la etapas posteriores. Llegados a puntos medios o avanzados de implementación cumplir un requerimiento mal enfocado lastraba de forma severa el cumplimiento de la planificación. Han sido, por tanto, necesarias muchas modificaciones que, siendo sencillas, implicaban revisar de forma muy laboriosa fases anteriores.

La implementación ha sido muy costosa, porque no se disponía de formación previa en el uso de este tipo de entorno de trabajo. Por otra parte, aunque se han compartimentado todo lo posible las funcionalidades, la estrecha vinculación de unas con otras complicaba de forma reiterada la consecución de los objetivos planificados.

No obstante este tipo de problemas son también inherentes a la inexperiencia y es plausible pensar que se pueden adquirir cotas muy superiores de eficiencia para futuros desarrollos.

### 8.2. ALGUNAS PROPUESTAS DE MEJORA

La elaboración de este tipo de trabajos deja siempre numerosos puntos pendientes de mejora o como mínimo de consideración especial. El presente trabajo un mínimo un punto de inicio, pero puede asumir siempre nuevas cotas.

Algunas de las mejoras que se podrían hacer son:

- Establecer procedimientos de alta y baja de miembros que permitan conocer cuales son los datos necesarios para poder fabricar estadísticas más consistentes.  
Este punto se puede observar al elaborar la estadística número 5, que para un año no permite conocer los miembros sólo para ese año.
- Establecer de forma rigurosa que estados aplicables a los regalos tienen posibilidad de retorno, de forma tal que un regalo pagado si pueda volver a estado de añadido.
- Mejorar la tabla transacciones y su entorno para que permita guardar de forma mucho más amplia los datos relativos a los pagos efectuados y las particularidades de cada método.
- Romper la restrictiva política que liga un artículo a proveedor concreto, de forma tal que se pueda asociar el artículo al proveedor que ofrezca mejores condiciones.
- Establecer políticas de bonificación por consumo, fidelidad o pronto pago de un regalo, para que los regalos tengan más probabilidad de éxito en cuanto a su recaudación de fondos.

## **9. BIBLIOGRAFÍA**

Documentos de consulta:

- Plan docente de TFC Bases de datos relacionales
- Enunciado de TFC Bases de datos relacionales
- Bases de datos – Universitat Oberta de Catalunya
- Bases de datos II –Universitat Oberta de Catalunya
- Ingeniería del Software – Universitat Oberta de Catalunya

Enlaces web de interés:

- <http://es.wikipedia.org/wiki/PL/SQL>
- <http://www.devjoker.com/gru/tutorial-PL-SQL/PLSQ/Tutorial-PL-SQL.aspx>
- <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/107>
- [http://apuntesduoc.pbworks.com/w/file/54222708/040628\\_PLSQL\\_Basico.pdf](http://apuntesduoc.pbworks.com/w/file/54222708/040628_PLSQL_Basico.pdf)

## 10. ANEXO

### TABLAS DEL MÓDULO PRINCIPAL

Categoría		
ATRIBUTO	TIPO	OBLIGATORIO
id_categoria	NUMBER(8)	NOT_NULL
nombre	VARCHAR2(20)	NOT_NULL
descripción	VARCHAR2(100)	
Clave Primaria	PK_Categoría (id_categoria)	
Restricción	UK_Categoría UNIQUE (nombre)	

Proveedor		
ATRIBUTO	TIPO	OBLIGATORIO
id_proveedor	NUMBER(8)	NOT_NULL
nombre	VARCHAR2(20)	NOT_NULL
Clave Primaria	PK_Proveedor (id_proveedor)	
Restricción	UK_Proveedor UNIQUE (nombre)	

Artículo		
ATRIBUTO	TIPO	OBLIGATORIO
id_artículo	NUMBER(8)	NOT_NULL
categoría	NUMBER(8)	NOT_NULL
descripción	VARCHAR2(40)	NOT_NULL
precio	NUMBER(12,2)	NOT_NULL
IVA	NUMBER(2)	NOT_NULL
proveedor	NUMBER(8)	NOT_NULL
Clave Primaria	PK_Artículo (id_artículo)	
Clave Foránea	FK(categoría) REFERENCES Categoría(id_categoria)	
Clave Foránea	FK(proveedor) REFERENCES Proveedor (id_proveedor)	
Restricción	CK_IVA → Posible valores: 4, 10, 21	

Miembro		
ATRIBUTO	TIPO	OBLIGATORIO
id_miembro	NUMBER(8)	NOT_NULL
email	VARCHAR2(30)	NOT_NULL
nombre	VARCHAR2(20)	NOT_NULL
apellido1	VARCHAR2(20)	NOT_NULL
apellido2	VARCHAR2(20)	NOT_NULL
dirección	VARCHAR2(40)	NOT_NULL
ciudad	VARCHAR2(40)	NOT_NULL
país	VARCHAR2(40)	NOT_NULL
Clave Primaria	PK_Artículo (id_miembro)	
Restricción	UK_Miembro UNIQUE (email)	

Desea		
ATRIBUTO	TIPO	OBLIGATORIO
id_deseo	NUMBER(8)	NOT NULL
homenajead	NUMBER(8)	NOT NULL
regalo	NUMBER(8)	NOT NULL



**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

fecha_alta	DATE	NOT NULL
Clave Primaria	PK_Desea (id_deseo)	
Clave Foránea	FK(homenajeado) REFERENCES Miembro(id_miembro)	
Clave Foránea	FK(regalo) REFERENCES Artículo (id_artículo)	
Restricción	UK_Desea UNIQUE (homenaje,artículo)	

Grupo		
ATRIBUTO	TIPO	OBLIGATORIO
id_grupo	NUMBER(8)	NOT NULL
homenajeado	NUMBER(8)	DEFAULT '0'
receptor	NUMBER(8)	DEFAULT '0'
dedicatoria_grupal	VARCHAR2(60)	NOT NULL
Clave Primaria	PK_Grupo (id_grupo)	
Clave Foránea	FK(homenajeado) REFERENCES Miembro(id_miembro)	

Agrupa		
ATRIBUTO	TIPO	OBLIGATORIO
email	VARCHAR2(30)	NOT NULL
grupo	NUMBER(8)	NOT NULL
dedicatoria_grupal	VARCHAR2(40)	
Clave Primaria	PK_Agrupa (email, grup)	
Clave Foránea	FK(email) REFERENCES Miembro(email)	
Clave Foránea	FK(grupo) REFERENCES Grupo(id_grupo)	

Regala		
ATRIBUTO	TIPO	OBLIGATORIO
id_regalo	NUMBER(8)	NOT NULL
grupo	NUMBER(8)	NOT NULL
deseo	NUMBER(8)	NOT NULL
fecha_inicio_recaudación	DATE	NOT NULL
fecha_final_recaudación	DATE	NOT NULL
estado_regalo	VARCHAR2(30)	DEFAULT 'AFEGIT_A_UN GRUP' NOT NULL
Clave Primaria	PK_Regala(id_regalo)	
Clave Foránea	FK(grupo) REFERENCES Grupo(id_grupo)	
Clave Foránea	FK(deseo) REFERENCES Grupo(id_deseo)	
Restricción	UK_Regala UNIQUE (grupo, deseo)	
Restricción	CK_Regala fecha_final_recaudación>fecha_inicio_recaudación	
Restricción	CK_Estado_regalo→ Posible valores: 'AFEGIT_A_UN GRUP', 'ELIMINAT', 'NO_SUFICIENTS_FONS', 'PAGAMENT_COMPLETAT', 'ENVIAT', 'REBUT'	

Método		
ATRIBUTO	TIPO	OBLIGATORIO
id_método	NUMBER(8)	NOT NULL
método	VARCHAR2(50)	NOT NULL
Clave Primaria	PK_Método(id_método)	

Transacción		
ATRIBUTO	TIPO	OBLIGATORIO
id_transacción	NUMBER(8)	NOT NULL
regalo	NUMBER(8)	NOT NULL
aportador	NUMBER(8)	NOT NULL
cantidad	NUMBER(8)	NOT NULL

**DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS PARA LA GESTIÓN DE REGALOS GRUPALES DE UNA RED SOCIAL**

tipo	VARCHAR2(30)	NOT NULL
traza_abono	NUMBER(8)	NULL
fecha	DATE	NOT NULL
forma_pago	NUMBER(8)	NOT NULL
visible	VARCHAR2(30)	NOT NULL
Clave Primaria	PK_Transacción (id_transacción)	
Clave Foránea	FK(regalo) REFERENCES Regala(id_regalo)	
Clave Foránea	FK(aportador) REFERENCES Miembro(id_miembro)	
Clave Foránea	FK(forma_pago) REFERENCES Método(id_método)	
Restricción	CK_tipo → Posible valores: 'PAGO','ABONO'	
Restricción	CK_visible → Posible valores: 'SI','NO'	

EST1_Veces_Regalado		
ATRIBUTO	TIPO	OBLIGATORIO
producto	NUMBER(8)	NOT NULL
año	NUMBER(4)	NOT NULL
numero_veces	NUMBER(4)	NOT NULL
Clave Primaria	PK_EST1_Veces_Regalado (producto, año)	

EST2_Total_Regalos		
ATRIBUTO	TIPO	OBLIGATORIO
año	NUMBER(4)	NOT NULL
total_regalos	NUMBER(8)	NOT NULL
Clave Primaria	PK_EST2_Total_Regalos (año)	

EST2_Total_Regalos		
ATRIBUTO	TIPO	OBLIGATORIO
año	NUMBER(4)	NOT NULL
total_regalos	NUMBER(8)	NOT NULL
Clave Primaria	PK_EST2_Total_Regalos (año)	