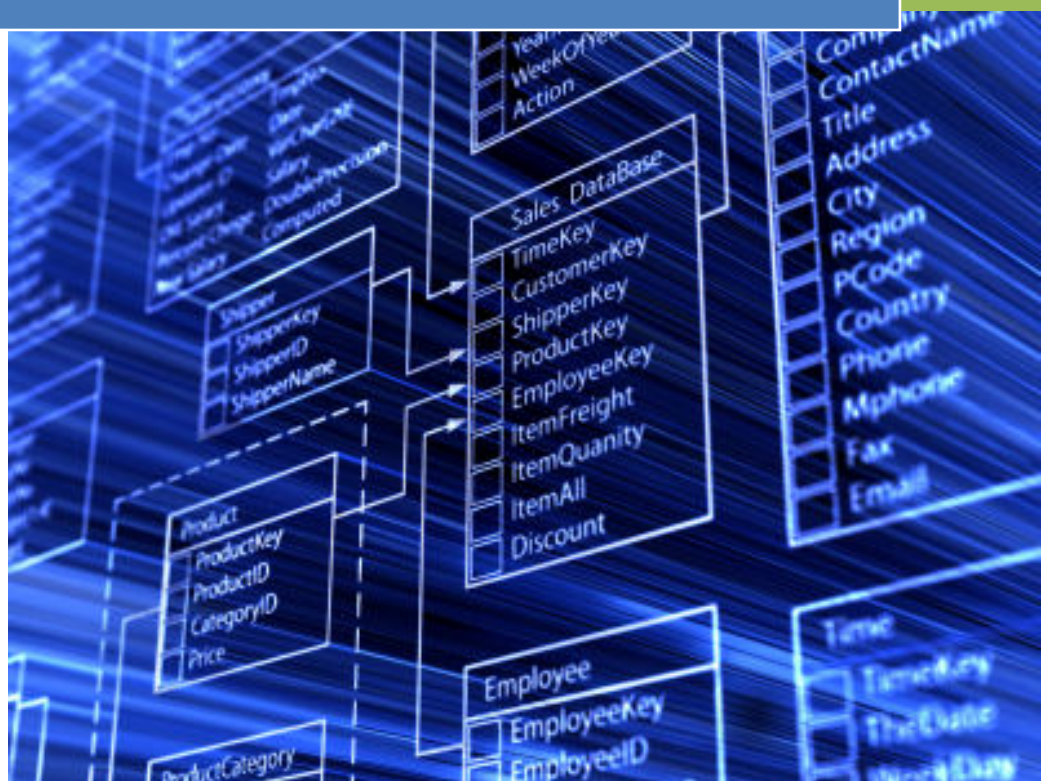


INGENIERÍA TÉCNICA EN INFORMÁTICA DE GESTIÓN

MEMORIA

TFC



Nombre Estudiante

José Luis Sallent Sánchez

Nombre Consultor

Juan Martínez Bolaños

11 de Enero de 2016

***A mis hijos, Daniel y Anna
y en especial a mi mujer, Marta,
por el apoyo incondicional
que me han dado estos años***

RESUMEN

Este documento forma parte del Trabajo de Fin de Carrera (TFC) en el área de Bases de Datos y en el marco de los estudios de Ingeniería Técnica en Informática de Gestión. El contenido de esta asignatura se centra en la organización, gestión y desarrollo de un proyecto informático desde un punto de vista práctico. Concretamente en este semestre se pide la creación de una Base de Datos dados unos requisitos iniciales expuestos en un enunciado.

Al abordar un proyecto de estas características hay que enfrentarse a 4 fases bien diferenciadas: el inicio, la calificación, el desarrollo y el cierre. Tomando como referencia el texto recibido como enunciado, se puede determinar que existe un objetivo consistente en consolidar los conocimientos adquiridos en los estudios de Ingeniería Técnica en Informática de Gestión y que se puede tomar como punto de inicio.

A lo largo del semestre Septiembre 2015 – Enero 2016 he ido llevando a cabo las 3 fases siguientes y como resultado he creado una Base de Datos Relacional usando Oracle Express 11g. La presente Memoria pretende sintetizar todo el trabajo realizado, desde la planificación y el control del mismo hasta las distintas etapas del desarrollo y las pruebas finales.

INDICE

1. Introducción	4
2. Definición del proyecto.....	5
2.1 Descripción.....	5
2.2 Objetivos del proyecto	5
2.3 Presentación de los recursos disponibles	6
2.4 Metodología para el desarrollo de la Base de Datos.....	6
2.5 Programación temporal.....	9
2.6 Asignación de recursos.....	12
2.7 Plan de contingencia	13
2.8 Comparativa resultado final con planificación	14
2.9 Productos obtenidos	15
3. Análisis de requerimientos	17
3.1 Descripción.....	17
3.2 Casos de uso.....	17
3.3 Requerimientos funcionales.....	22
3.4 Requerimientos no funcionales.....	30
4. Diseño de la base de datos.....	31
4.1 Diseño conceptual	31
4.2 Diseño lógico	33
4.3 Diseño físico	34
4.3.1 Características Oracle Express 11g.....	34
4.3.2 Nomenclatura	36
4.3.3 Diseño de relaciones	37
4.3.4 Definición de reglas de la BD.....	37
5. Estructura de la aplicación	39
5.1 Estructura general	39
5.2 Lógica interna	39
6. Pruebas	43
7. Valoración económica.....	52
8. Conclusiones	55
9. Glosario	56
10. Bibliografía.....	57
11. Anexo I. Correspondencia con el cliente.....	58

INDICE DE TABLAS

Figura 1: Etapas del ciclo de vida en cascada	7
Figura 2: Etapas del ciclo de vida adaptado al proyecto	8
Figura 3: Duración estimada del proyecto por fases	9
Figura 4: Duración estimada fase 1	10
Figura 5: Duración estimada fase 2	10
Figura 6: Duración estimada fase 3	11
Figura 7: Duración estimada fase 4	11
Figura 8: Diagrama de Gantt del proyecto	12
Figura 9: Porcentaje de horas disponibles no consumidas.....	13
Figura 10: Comparación horas planificadas con reales PAC2.....	14
Figura 11: Comparación horas planificadas con reales PAC3.....	15
Figura 12: Documentos a entregar.....	16
Figura 13: Diagrama de casos de uso.	18
Figura 14: Diagrama E/R.	31
Figura 15: Oracle Database XE Requerimientos para Microsoft Windows 32-bit	35
Figura 16: Estructura general del código.....	39
Figura 17: Funcionamiento general interno de un procedimiento.....	40
Figura 18: Estructura PKG_EST_DIR.....	42
Figura 19: Resultado pruebas PKG_ADMINISTRACIÓN	44
Figura 20: Resultado pruebas PKG_VENTAS	45
Figura 21: Resultado pruebas PKG_EST_DIR.....	45
Figura 22: Resultado pruebas PKG_CONSULTAS.....	46
Figura 23: Resultado pruebas PKG_CONTROL	46
Figura 24: Coste hora por perfil	52
Figura 25: Consumo de horas planificadas según perfil.....	52
Figura 26: Coste total planificado de horas empleadas.....	53
Figura 27: Determinación del precio de venta inicial	53
Figura 28: Consumo de horas reales según perfil.....	54
Figura 29: Coste total real de horas empleadas.....	54
Figura 30: Determinación del precio de venta final	54

1. Introducción

Todo proyecto de desarrollo de software, una vez fijada de un modo u otro la necesidad de llevarlo a cabo, necesita de una buena planificación y un posterior control que permita tomar las decisiones adecuadas ante las desviaciones que se pudieran producir. En el capítulo siguiente se expone la que se conoce como fase de calificación en la que se describe el proyecto, se fijan objetivos, se planifican las actividades, etc. En el mismo capítulo también se detallan las desviaciones detectadas sobre la planificación y cómo éstas han sido corregidas. Siendo el análisis de desviaciones un apartado más propio del cierre que de la calificación, se ha considerado su inclusión en ese capítulo para relacionarlo más fácilmente con la planificación.

Cuando ya se tiene claro lo que hay que hacer y cómo, se pueden acometer el resto de fases. Como se expondrá y justificará en el capítulo segundo, se ha optado por un ciclo de desarrollo en cascada. Por tanto el resto de capítulos seguirá la estructura lógica de dicho ciclo de vida.

Por lo que se refiere al análisis de requerimientos, hay que considerar que existen muchas maneras de reflejarlos en un documento. En este caso he optado por plantear primero un diagrama de casos de uso. Esto va a permitir que los desarrolladores entiendan mejor la aplicación en caso de que posteriormente haya que hacer una interfaz gráfica para los usuarios. Posteriormente se listan los requerimientos funcionales y no funcionales de la Base de Datos a la vez que se plantean propuestas de mejora. Algunas de estas propuestas han sido aceptadas por el cliente y otras no. De esto se deja constancia a lo largo del texto en capítulo tercero.

En el apartado dedicado al desarrollo de la Base de Datos que se expone en el capítulo cuarto, se trata en detalle la formulación de un modelo Entidad – Relación que va a permitir diseñar una Base de Datos que satisfaga todos los requerimientos expuestos sin considerar aún el lenguaje de programación. En el mismo capítulo se hará la transformación de ese diseño conceptual a un diseño lógico considerando que se usará una base de datos relacional. En lo que se refiere al diseño físico no se abordarán los aspectos relativos al almacenamiento físico de los datos, caminos de acceso, etc. Se entiende que Oracle Express es una herramienta lo suficientemente potente como para dar respuesta a las necesidades que en este sentido que pueda requerir la aplicación.

La parte de programación será entregada con la aplicación pero se expone brevemente en el capítulo quinto su estructura interna y la de cada paquete que la compone lo que permitirá un mejor entendimiento de la lógica de funcionamiento. En el capítulo sexto se describirá en detalle las pruebas realizadas y su resultado.

Para finalizar se dedica un capítulo para apuntar brevemente las conclusiones que se pueden extraer del trabajo realizado.

2. Definición del proyecto

2.1 Descripción

El proyecto consiste en realizar las tareas necesarias de planificación, control y producción para la creación de una Base de Datos según un enunciado dado en el que se fijan unos requisitos iniciales. En el capítulo tercero, se hará un análisis exhaustivo de dichos requisitos. En este capítulo sólo se hará referencia a los objetivos que debe cumplir y que se tratan en el apartado 2.2.

El enunciado recibido hace referencia a la necesidad de una cadena de ropa de tener un mecanismo que permita a la dirección de la compañía hacer una serie de consultas rápidas sobre el funcionamiento del negocio. Para ello ha abierto un concurso con el ánimo de recibir las propuestas de diferentes proveedores.

Para el desarrollo habrá que usar Oracle Express como Sistema Gestor de Bases de Datos por imposición del enunciado. En consecuencia se descarta desde el inicio cualquier actividad encaminada a seleccionar un Sistema Gestor diferente y se deja ese aspecto fuera del alcance del trabajo.

2.2 Objetivos del proyecto

El objetivo principal de este trabajo consiste en consolidar los conocimientos adquiridos durante los estudios de Ingeniería Técnica en Informática de Gestión. Dada el área en la que se enfoca el trabajo, en particular se centra en las asignaturas relacionadas con Bases de Datos.

Dicho objetivo principal se puede concretar en:

- Poner en práctica los conocimientos adquiridos en las asignaturas Bases de Datos I, Bases de Datos II e Ingeniería del Software.
- Utilizar el lenguaje PL/SQL y SQL Dinamic.
- Ampliar conocimientos usando nuevas herramientas.

En particular, para conseguir dichos objetivos se marca como objetivo secundario el diseño de una Base de Datos para una cadena de ropa que permita almacenar:

- Número de ventas realizadas diariamente por cada uno de los productos disponibles en el catálogo.
- Información estadística accesible en tiempo real para la dirección de la cadena que le permitirá tomar decisiones sobre el negocio.
- Información relativa a las tiendas de la cadena.

2.3 Presentación de los recursos disponibles

Este proyecto se ha realizado de forma individual por el alumno contando con la ayuda del consultor de la asignatura para la que se desarrolla.

El autor es Licenciado en Administración y Dirección de Empresas y cuenta con más de 15 años de experiencia en el ámbito financiero de empresas de tamaño medio en un entorno internacional. Así mismo, ha trabajado en ocasiones en las etapas finales de implantación de sistemas ERP (*Enterprise Resource Planning*) estándar participando en las cargas de datos y los test de funcionamiento. Además, ha desarrollado en VBA (*Visual Basic for Applications*) aplicaciones financieras y de reporte de uso interno en Excel en las organizaciones para las que ha trabajado.

Actualmente está cursando esta asignatura como último paso para la consecución del título de Ingeniero Técnico en Informática de Gestión. Por tanto tiene conocimientos previos en Bases de Datos al haber cursado Bases de Datos I y II así como Ingeniería del Software y Gestión y Organización de proyectos informáticos.

Para la realización de este trabajo se cuenta con los siguientes medios técnicos:

Hardware:

- Ordenador HP ProBook 640 G1, Intel Core i3 4000M CPU @2.40 GHz 2,40 GHz, 4 GB RAM, equipado con Windows 7 y Microsoft Office.
- Ordenador Asus, Intel Core i5 4210U CPU @ 1.70 GHz 2,40GHz, 4 GB RAM, equipado con Windows 8.1 y Microsoft Office.

Software:

- Gantt Project 2.6.6 Brno (build 1715).
- Oracle Database XE 11g.
- Oracle Sql Developer 4.1.1.
- Paquete de ofimática, Microsoft Office 2007.
- Navegadores web, Internet Explorer y Google Chrome.

Se dispone también de conexión a Internet y acceso al campus virtual.

2.4 Metodología para el desarrollo de la Base de Datos

Existen diferentes ciclos de vida en el desarrollo de software. De entre todos ellos, en este trabajo se va a seguir el denominado ciclo de vida clásico o en cascada por los siguientes motivos:

- Es un método dirigido a obtener un resultado al final en cada etapa. Dado que durante el TFC hay que hacer entregas parciales se podrán ir entregando los resultados de las diferentes etapas ya terminados.
- Se pueden detectar los errores a través de las entregas parciales. El coste de corrección por tanto es menor que si se detectan al final del proyecto.
- La planificación es más sencilla de realizar y de controlar.

El ciclo de vida clásico o en cascada se compone de las siguientes fases:

- Análisis previo: En esta etapa se definen los rasgos previos del software.
- Análisis de requisitos: Esta etapa está enfocada a definir con detalle los requisitos que debe satisfacer el software.
- Diseño: El diseño es el momento en el que se establece la solución necesaria para dar respuesta a los requisitos detectados en la etapa anterior.
- Programación: En este punto se traduce el diseño al lenguaje de programación.
- Prueba: Consiste en realizar pruebas al software generado.
- Mantenimiento: Una vez instalado el software hay que mantenerlo y corregirlo mejorando sus funciones y su eficacia.

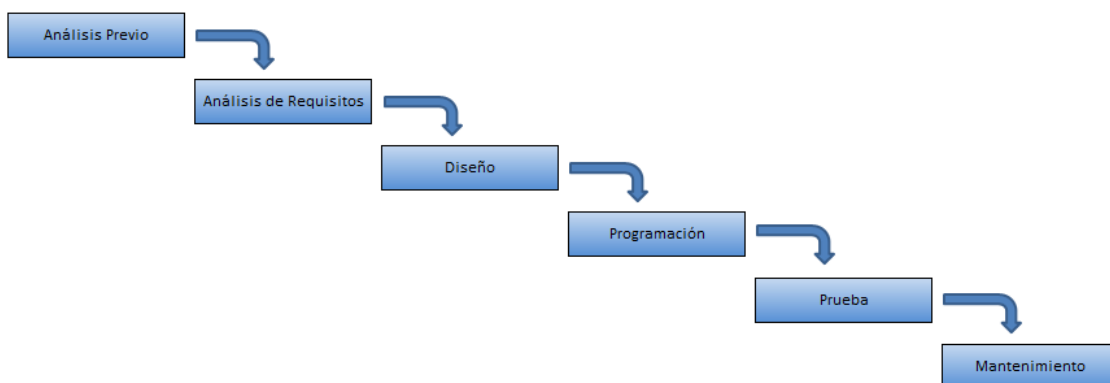


Figura 1: Etapas del ciclo de vida en cascada

Dado que el proyecto finaliza con la entrega del producto una vez hechas las pruebas no se trata en este trabajo la última etapa de mantenimiento.

En este ciclo de vida cada etapa es el comienzo de la siguiente y por tanto no se permite el regreso a etapas pasadas. Hay que indicar que frente a la rigidez de este método conviene permitir una cierta flexibilidad para ajustar algún elemento de las etapas anteriores. Sobre todo entre la etapa de pruebas y la de programación ya que las pruebas pueden detectar algunos errores en el código que habría que solucionar.

Por ello me he permitido crear una relación “hacia atrás” en la etapa de prueba con el único fin de hacer algún ajuste al código.

Si el software que se está desarrollando es una Base de Datos, en la etapa de diseño habrá que observar tres actividades:

- Diseño conceptual: Actividad destinada a formular la estructura de la futura Base de Datos sin considerar la tecnología que se va a emplear ni el tipo de Base de Datos ni el Sistema Gestor de Base de Datos. Tiene como resultado un modelo de datos en algún lenguaje de alto nivel. A los efectos de este trabajo se va a usar un modelo Entidad-Relación (E-R).
- Diseño lógico: En esta actividad se adapta el diseño conceptual a la tecnología que se va a emplear. Dado que utilizaré SQL Express como Sistema gestor de Base de Datos, en este trabajo convertiré el modelo E-R anterior en un conjunto de relaciones con atributos, claves, etc.
- Diseño físico: La última actividad del diseño está destinada a conseguir una mayor eficiencia. Se piden accesos rápidos a determinada información por lo que puede ser interesante buscar combinaciones de relaciones o campos calculados que permitan un acceso más rápido. No entraré sin embargo en los aspectos relativos al almacenamiento físico de los datos, caminos de acceso o tamaños de las memorias intermedias ya que excede el alcance del proyecto.

Conforme a lo expuesto anteriormente, gráficamente se pueden resumir las etapas que se van a seguir para la confección de la Base de Datos solicitada del siguiente modo:

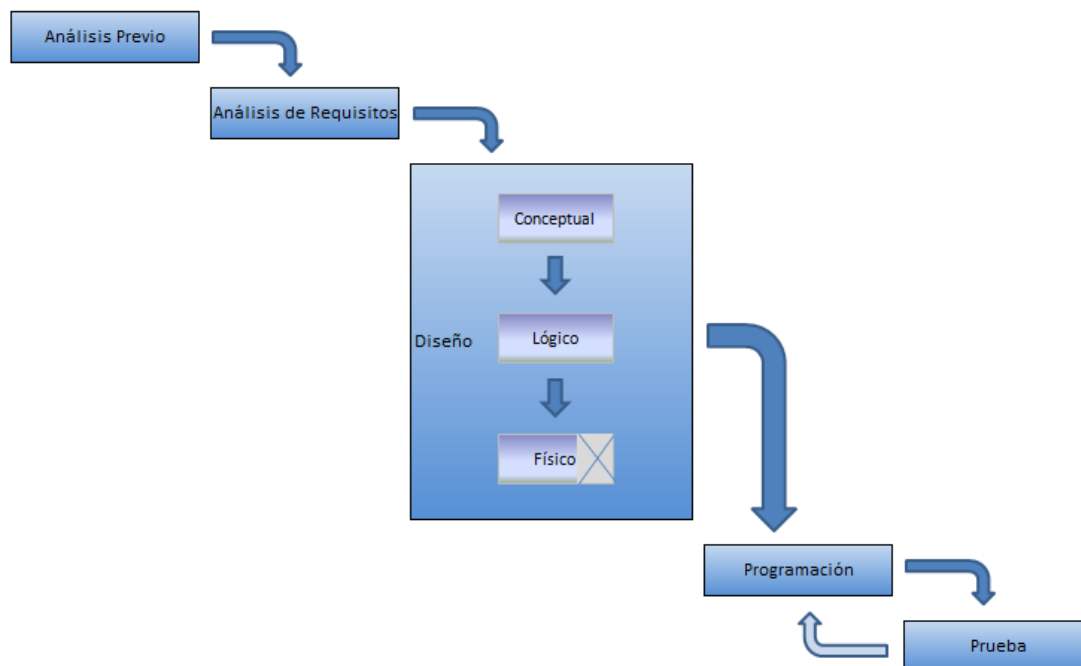


Figura 2: Etapas del ciclo de vida adaptado al proyecto

No obstante lo anterior, si estuviéramos en un caso de venta real, lo conveniente sería utilizar un ciclo de vida con prototipos. De ese modo no se invertiría en un desarrollo completo hasta tener claro que se ha ganado el contrato pues es un concurso abierto para el desarrollo de una Base de Datos específica de la que no se espera una venta al público en general. Además se podría presentar al cliente un tangible antes de iniciar un desarrollo por completo, lo que sin duda apoyaría la venta.

2.5 Programación temporal

El trabajo se va a desarrollar en 4 fases: El plan de trabajo, la PAC2, la PAC3 y la entrega final. Para cada una de las fases, menos para la primera, se ha establecido un punto de control para analizar entre el autor y el consultor la evolución y dudas sobre el trabajo que se desarrolla. El establecer ahora unos puntos de control no impide que se fijen otros si llegado el momento se considera necesario.

La duración estimada del proyecto es de 191 horas. Esto garantiza que se cumplen los 7,5 créditos de los que consta la asignatura. Establecido que cada crédito supone aproximadamente 25 horas de carga de trabajo, ésta habría de ser no inferior a 187,5 horas.

La primera de las etapas dará comienzo el 16 de Septiembre de 2015 y se ha de poder entregar la documentación final el 11 de Enero de 2016.

A continuación se presenta la duración estimada por fases.

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Descripción
1		PLAN DE TRABAJO	16-sep-15	05-oct-15	34	Consiste en la elaboración de la calificación del proyecto. Definición de objetivos y programación
2	1	PAC2	06-oct-15	09-nov-15	60,5	Análisis de requerimientos y diseño conceptual.
3	2	PAC3	10-nov-15	10-dic-15	52,5	Análisis lógico e implementación.
4	3	Memoria	11-dic-15	11-ene-16	44,5	Redacción de documentación final a entregar.

Figura 3: Duración estimada del proyecto por fases

Plan de trabajo.

En esta fase se van a definir el alcance, los objetivos del proyecto y la metodología a seguir para el desarrollo de la Base de Datos. Además se va a elaborar la planificación temporal de las tareas a realizar de modo que luego se pueda hacer un adecuado seguimiento. Comienza el 16 de Septiembre con la apertura de las aulas y termina el 5 de Octubre con la entrega del plan de trabajo.

MEMORIA - TFC – Bases de Datos

José Luis Sallent

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Descripción
1		PLAN DE TRABAJO	16-sep-15	05-oct-15	34	Consiste en la elaboración de la calificación del proyecto. Definición de objetivos y programación
1.1		Preparación y lectura	16-sep-15	18-sep-15	4	Descarga y lectura de caso práctico, recomendaciones y plan docente.
1.2	1.1	Planteamiento del proyecto, definición y planificación	19-sep-15	28-sep-15	14	Boceto de texto con ideas básicas a desarrollar. Planteamiento de actividades a ejecutar y su horizonte temporal. Determinación de fechas clave. Determinación de la metodología a seguir.
1.3	1.1, 1.2	Redacción de contenidos	19-sep-15	04-oct-15	7	Elaboración del documento a entregar correspondiente a la PAC1.
1.4		Preparación entorno de trabajo	16-sep-15	24-sep-15	8,5	Actividades enfocadas a tener todo el software listo antes de comenzar a trabajar en el análisis y desarrollo. Actividades de refresco de conocimientos
1.4.1		Lectura de apuntes asignaturas anteriores	16-sep-15	24-sep-15	6	Lectura de apuntes de BD I y II, de ingeniería del Software y de Gestión y Organización de Proyectos Informáticos.
1.4.2		Instalación Oracle Express, SQL Developer	18-sep-15	18-sep-15	2	Descarga, instalación y lectura de documentación de Oracle Express y SQL Developer.
1.4.3		Instalación Gantt Project	19-sep-15	19-sep-15	0,5	Descarga, instalación y lectura de documentación de Gantt Project.
1.5	1.3	Entrega del plan de trabajo	05-oct-15	05-oct-15	0,5	Entrega de PAC1 en el apartado de envíos del aula.

Figura 4: Duración estimada fase 1

PAC2.

Durante el periodo asignado a esta fase se trabajará sobre la determinación de los requerimientos y el diseño conceptual de la Base de Datos. Este diseño dará como resultado un esquema basado en un modelo E–R que servirá de base para diseño lógico posterior. Comienza el 6 de Octubre y termina el 9 de Noviembre con la entrega de la PAC2.

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Descripción
2	1	PAC2	06-oct-15	09-nov-15	60,5	Análisis de requerimientos y diseño conceptual.
2.1		Análisis de requerimientos	06-oct-15	11-oct-15	12	Lectura de los requerimientos del enunciado. Comunicación con el aula y el profesor para resolver dudas y matizar los requerimientos.
2.2	2.1	Redacción de documento de especificación	12-oct-15	13-oct-15	4	Redacción de un documento que recoja los requerimientos detectados.
2.3	2.1	Diseño conceptual	12-oct-15	01-nov-15	32	Elaboración del diseño E-R según los requerimientos detectados.
2.4	2.2, 2.3	Redacción de documento a entregar	02-nov-15	08-nov-15	12	Elaboración del texto a entregar de la PAC2
2.5	2.4	Entrega de PAC 2	09-nov-15	09-nov-15	0,5	Entrega de PAC2 en el apartado de envíos del aula.

Figura 5: Duración estimada fase 2

PAC3.

La tercera fase del trabajo estará dedicada al diseño lógico de la Base de Datos así como a la programación y pruebas de la misma. Comienza el 10 de Noviembre y termina el 10 de Diciembre con la entrega de la PAC3.

MEMORIA - TFC – Bases de Datos

José Luis Sallent

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Descripción
3	2	PAC3	10-nov-15	10-dic-15	52,5	Análisis lógico e implementación.
3.1		Diseño lógico	10-nov-15	15-nov-15	12	Transformación del modelo ER a un diseño lógico para BD relacional.
3.2	3.1	Programación de Base de datos	16-nov-15	25-nov-15	20	Programación de la base de Datos formulada en el diseño lógico. Programación de procedimientos y disparadores.
3.3	3.2	Redacción de documento a entregar	26-nov-15	03-dic-15	6	Traspasar en un documento las tablas generadas en el diseño lógico y los scripts generados.
3.4	3.2	Elaboración de los grupos de pruebas	26-nov-15	29-nov-15	6	Prepara los grupos de pruebas para probar la base de datos así como la documentación que habrá de generarse.
3.5	3.4	Ejecución de las pruebas y retoques de código	30-nov-15	03-dic-15	4	Lanzamiento de las pruebas y retoques en el código de la Base de Datos si fuera necesario.
3.6	3.2, 3.5	Preparar documentos final para entrega	04-dic-15	09-dic-15	4	Revisión de la documentación generada y revisión final del documento a entregar.
3.7	3.6	Entrega PAC 3	10-dic-15	10-dic-15	0,5	Entrega de PAC3 en el apartado de envíos del aula.

Figura 6: Duración estimada fase 3

ENTREGA FINAL.

En esta última fase se va a elaborar la Memoria, la presentación virtual y se va a preparar todo lo necesario para la entrega final. Debe servir para dar una forma profesional al documento y para preparar una exposición explicativa de los resultados del proyecto. Comienza el 11 de Diciembre de 2015 y termina el 11 de Enero de 2016 con la entrega de la Memoria, la presentación y el código elaborado para la Base de Datos.

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Descripción
4	3	Memoria	11-dic-15	11-ene-16	44,5	Redacción de documentación final a entregar.
4.1		Determinar el tipo de licencia	11-dic-15	11-dic-15	2	Determinación del tipo de licencia para la publicación del TFC.
4.2		Preparación de la memoria	12-dic-15	07-ene-16	17	Preparación del documento de Memoria tomando como base los documentos entregados en PAC1, PAC2 y PAC3.
4.3		Preparación de la presentación	12-dic-15	07-ene-16	17	Preparación de una presentación resumen del trabajo realizado.
4.4		Preparación del paquete con el código a entregar	12-dic-15	07-ene-16	6	Preparación de los ficheros con el código a entregar.
4.5		Revisión del documento a entregar	08-ene-16	10-ene-16	2	Revisión de toda la documentación a entregar.
4.6		Envío de la memoria	11-ene-16	11-ene-16	0,5	Entrega de la memoria, código programado y presentación virtual en el apartado de envíos del aula.

Figura 7: Duración estimada fase 4

Seguidamente se muestra en un diagrama de GANTT las diferentes actividades que conforman el proyecto.

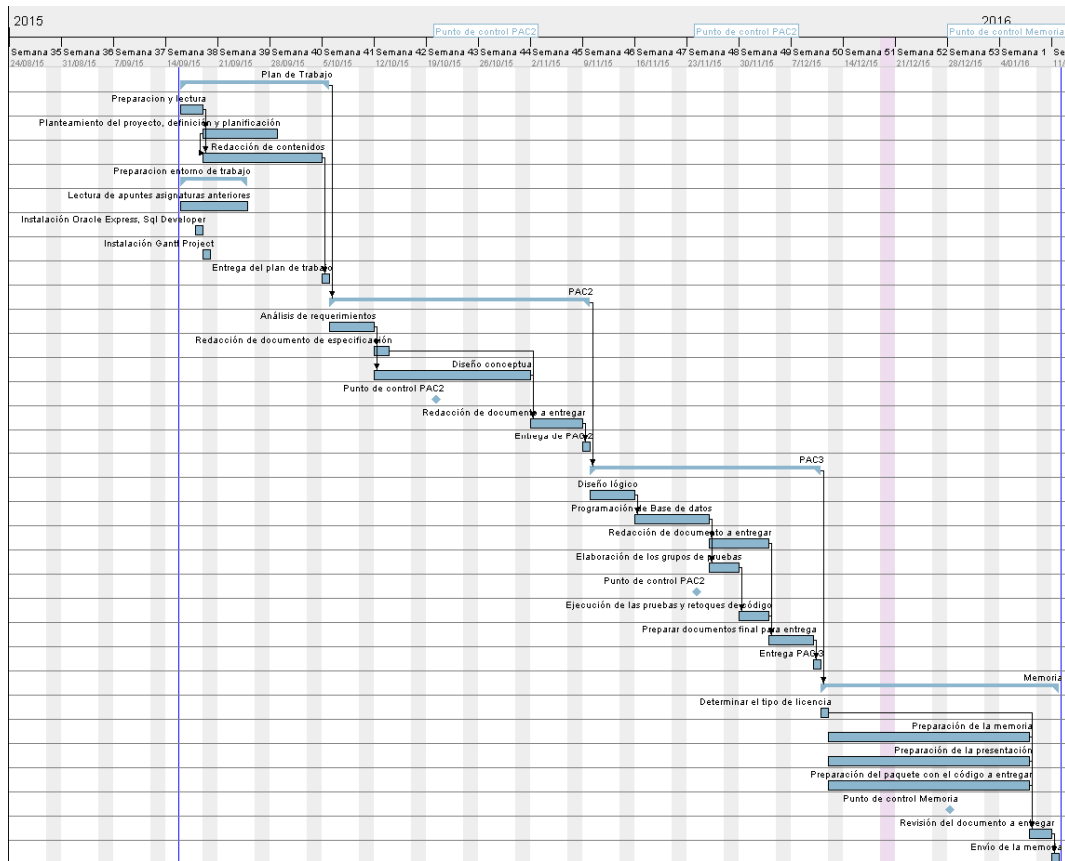


Figura 8: Diagrama de Gantt del proyecto

2.6 Asignación de recursos

Como se ha visto en el punto anterior, el proyecto se desarrolla desde el 16 de Septiembre de 2015 hasta el 11 de Enero de 2016. De los días que transcurren entre el inicio y el final hay que restar 4 días, aún por determinar, en el periodo navideño y en los que no se podrá trabajar debido a viajes personales. Ya que no se consideran fines de semana ni festivos, hay un total de 114 días hábiles.

Debido a las obligaciones laborales no se pueden establecer jornadas típicas de 8 horas diarias, 40 semanales. Se establece una jornada aproximada de 2 horas diarias, 7 días a la semana. Este reparto no necesariamente implica 2 horas exactas cada día si no que lo normal será que durante los días laborales se hagan menos horas y se recuperen el fin de semana, completando cada Domingo la jornada semanal.

Por tanto, en total hay disponibles 228 horas lo que supone 37 horas de más sobre el tiempo estimado de desarrollo del proyecto. Esto deja un margen de unas dos horas semanales no comprometidas que serán aplicadas para el plan de contingencia expuesto en el punto 2.7.

2.7 Plan de contingencia

Se han detectado las siguientes áreas de riesgo:

- **Salud, enfermedad leve:** en caso de enfermedad leve se perderían de dos a tres días. Serían unas 6 horas como máximo. Riesgo bajo.

Plan: Horas recuperables en fin de semana.

- **Familiar:** Los retrasos recogidos en esta área de riesgos se pueden producir por la enfermedad de algún miembro de la familia que en un momento puntual no permita dedicar el tiempo requerido o por actividades escolares o extraescolares de los hijos que requieran mayor dedicación de la estimada. Riesgo medio.

Plan: Horas recuperables en fin de semana.

- **Laboral:** Pueden surgir situaciones que exijan una mayor dedicación al trabajo. Ocurre con frecuencia. Riesgo alto.

Plan: Debido a la posición que ocupa en la empresa en la que trabaja, el alumno dispone de cierta flexibilidad lo que permitirá tomar algún día, pasado el punto de exceso de trabajo, para recuperar el tiempo perdido.

- **Técnico:** Fallos de hardware y software. Riesgo bajo.

Plan: Se trabaja con un ordenador con una antigüedad de dos meses por lo que no se estima que se averíe. De todos modos se dispone de un segundo ordenador con la misma antigüedad. Se hacen copias de seguridad semanales en un disco duro externo.

Visto que el proyecto requiere un consumo de 191 horas y que hay disponibles 228 que no serán asignadas a otras actividades, queda un margen de 37 horas (mínimo) para afrontar imprevistos. Este margen de horas se distribuye en las distintas fases del siguiente modo:

Id	Previa	Tarea	Fecha inicio	Fecha fin	Horas	Horas disponibles	Horas no aplicadas	% ajuste
1		PLAN DE TRABAJO	16-sep-15	05-oct-15	34	40	6	15%
2	1	PAC2	06-oct-15	09-nov-15	60,5	70	9,5	14%
3	2	PAC3	10-nov-15	10-dic-15	52,5	62	9,5	15%
4	3	Memoria	11-dic-15	11-ene-16	44,5	56	11,5	21%
Total					191,5	228	36,5	16%

Figura 9: Porcentaje de horas disponibles no consumidas

Con esta distribución y como se puede apreciar en la tabla anterior, quedan más horas disponibles en la última fase del proyecto que en el resto de fases. Esto es positivo fundamentalmente por dos motivos:

- Siendo fechas navideñas es más probable que haya días en los que, incluso estando previsto, no se pueda trabajar.
- Si por algún motivo se producen retrasos a lo largo del proyecto que no haya sido posible recuperar con la aplicación del plan de contingencia se puede ajustar en la fase final.

Así mismo, conviene reseñar que las 228 horas reservadas para este proyecto pueden incrementarse en caso necesario.

2.8 Comparativa resultado final con planificación

Durante la calificación del proyecto estimé una carga de trabajo para la PAC2 de 60,5 horas. Si comparamos las horas reales con las previstas (figura 10) se observa que se ha producido una variación positiva de 5 horas en las actividades planificadas.

Tarea	Horas Plan.	Horas Real.	Dif.
Análisis de requerimientos	12	21	9
Redacción de documento de especificación	4	4	0
Diseño conceptual	32	17	-15
Diseño lógico	12	3	-9
Redacción de documento a entregar	12	10	-2
Entrega de PAC 2	0,5	0,5	0

Figura 10: Comparación horas planificadas con reales PAC2

La carga de trabajo estimada para el diseño conceptual se sobreestimó y llevó bastante menos tiempo del esperado. En total, en las actividades fijadas para la PAC2, se produjo una variación positiva de 8 horas. Esto permitió avanzar en la primera de las tareas fijadas para la PAC3, el diseño lógico, que también consumió menos tiempo del estimado.

El mayor consumo de tiempo en el análisis de requerimientos es debido a que al hacer la estimación no consideré abordar el estudio de casos de uso que finalmente incluí. Por otra parte, el diseño conceptual ha resultado más sencillo de lo que cabría esperar. En consecuencia el diseño lógico también resulta de más fácil elaboración y ambos han consumido menos tiempo del esperado.

También detecté que faltaba por planificar la actividad para diseño físico y decidí incluirla en la PAC3.

Durante la calificación del proyecto estimé una carga de trabajo para la PAC3 de 52,5 horas. Tras concluir las actividades planificadas en este periodo se observa que se ha producido una variación muy significativa de 58 horas (figura 11). Esto supone un 110% más de lo esperado. Revisando los consumos de horas por tareas observé que las mayores desviaciones se produjeron en las actividades de programación y de retoques de código tras la ejecución de pruebas.

Tarea	Horas Plan.	Horas Real.	Dif.
PAC3	52,5	110,5	58
Diseño físico	12	13,5	1,5
Programación de Base de datos	20	68,5	48,5
Redacción de documento a entregar	6	6	0
Elaboración de los grupos de pruebas	6	4	-2
Ejecución de las pruebas y retoques de código	4	12	8
Preparar documentos final para entrega	4	6	2
Entrega PAC 3	0,5	0,5	0

Figura 11: Comparación horas planificadas con reales PAC3

Al margen de que subestimé las horas destinadas a estas actividades, hay que destacar que al menos 24 horas de trabajo en programación han sido usadas en el intento de formular consultas sin el uso de los campos de precio unitario de venta y beneficio unitario de venta. Visto ese consumo excesivo busqué otra solución al diseño físico para cumplir con los requerimientos respecto a las estadísticas (ver apartado 4.3.3).

Para poder hacer frente a este exceso de horas sobre lo planificado se ha hecho uso del paquete de 9,5 horas disponibles no aplicadas (ver punto 2.7) para PAC3. El resto han sido obtenidas incrementando las horas dedicadas en determinados días.

En consecuencia se han consumido un total de 244,5 horas, incluidas las fases de calificación y entrega final. Esto supone un incremento de 53 horas, un 27,68%, sobre el tiempo estimado inicialmente.

2.9 Productos obtenidos

Cada una de las fases intermedias (Plan de Trabajo, PAC2 y PAC3) tiene su fin con la entrega de un documento donde se refleja en estado del trabajo realizado hasta ese momento. La última fase tiene como punto final la entrega de tres documentos donde se refleja todo el trabajo realizado durante el proyecto.

A continuación se resumen los documentos a entregar:

Fase	Documento	Nombre	Contenido
PLAN DE TRABAJO	Planificación	jlsallent_PlaTrebll.pdf	Calificación de proyecto.
PAC2	PAC2	jlsallent_PAC2.pdf	Requerimientos y diseño conceptual.
PAC3	PAC2	jlsallent_PAC3.pdf	Diseño lógico, código base de datos, paquetes de pruebas y resultados de test.
Entrega Final	Memoria	jlsallent_memoria.pdf	Entregas anteriores y conclusiones.
	Presentación	jlsallent_presentacio.pdf	Resumen en diapositivas del trabajo.
	Producto	jlsallent_producto.zip	Fuente y un fichero .txt donde se expliquen los pasos de instalación.

Figura 12: Documentos a entregar

- Memoria: Es un documento donde se refleja el trabajo realizado incluyendo los aspectos relativos a la elaboración del TFC y al desarrollo de la Base de Datos. Ha de tener un máximo de 60 páginas más portada y dedicatoria (opcional).
 - o Portada con título, nombre del estudiante, titulación, nombre del consultor y fecha.
 - o Resumen de unas 250 palabras.
 - o Índice de contenidos e índice de figuras.
 - o Cuerpo con el desarrollo del proyecto.
 - o Glosario.
 - o Bibliografía.
 - o Anexos.

- Presentación virtual: Se trata de un documento que ha de recoger un resumen del trabajo llevado a cabo. No debe superar las 20 diapositivas.

- Producto: Código generado en la programación de la Base de Datos.

No se contempla la entrega de una interfaz gráfica para el usuario de la Base de Datos.

3. Análisis de requerimientos

3.1 Descripción

Como se ha indicado en la introducción, en este apartado se va a desarrollar un diagrama de casos de uso y se van a listar los requerimientos funcionales y no funcionales que debe tener la Base de Datos.

Sin entrar en mucho detalle, pues no es necesario, diré que un diagrama de casos de uso es una herramienta útil para mostrar gráficamente las funciones de un software desde el punto de vista de las interacciones que tiene con su entorno. En este análisis habrá que describir los actores y los casos de uso. Los actores son los grupos de papeles (personas, sistemas o máquinas) que interactúan con el sistema de la misma forma. Un caso de uso representa una interacción entre un actor y el sistema. Para hacer la representación gráfica se ha usado el programa ArgoUML.

Por lo que se refiere a los requisitos, éstos son simplemente la descripción en un lenguaje normal del comportamiento, propiedades y restricciones que debe tener el software que se está desarrollando. Hay dos tipos de requerimientos, los funcionales que describen lo que debe hacer el software y los no funcionales que son aquellos comportamientos no asociados a un caso de uso concreto.

3.2 Casos de uso

Con la finalidad de mostrar gráficamente las funciones que debe proporcionar la BD, a continuación se van a exponer brevemente los casos de uso. Primero se describen los actores que interactúan de algún modo con la BD y tras el diagrama de casos de uso se especifica, de forma detallada, cada uno de ellos.

Administradores.

Es el grupo de personas encargada de mantener los datos auxiliares necesarios para el buen funcionamiento de la aplicación. Estos datos son aquellos que se refieren a las tiendas y a los productos del catálogo. Además se encargarán del correcto mantenimiento de la aplicación y de la resolución de incidencias.

Vendedores.

Son los empleados de las diferentes tiendas de la cadena y cuando realizan una venta deben introducirla en el sistema.

Directivos.

Son los responsables de la toma de decisiones y necesitan consultar las estadísticas generadas para apoyarlas. Dado que son superiores jerárquicamente a los vendedores, se contempla que los directivos también puedan tomar el papel de vendedor. Este hecho se muestra en el diagrama a través de una relación de especialización entre ambos actores.

ERP de la compañía.

En el momento en que un vendedor registra una venta hay una serie de datos económicos los cuales deben ser proporcionados de forma automática por el ERP que sirve para la gestión de compañía.

En la siguiente figura podemos ver el diagrama de casos de uso:

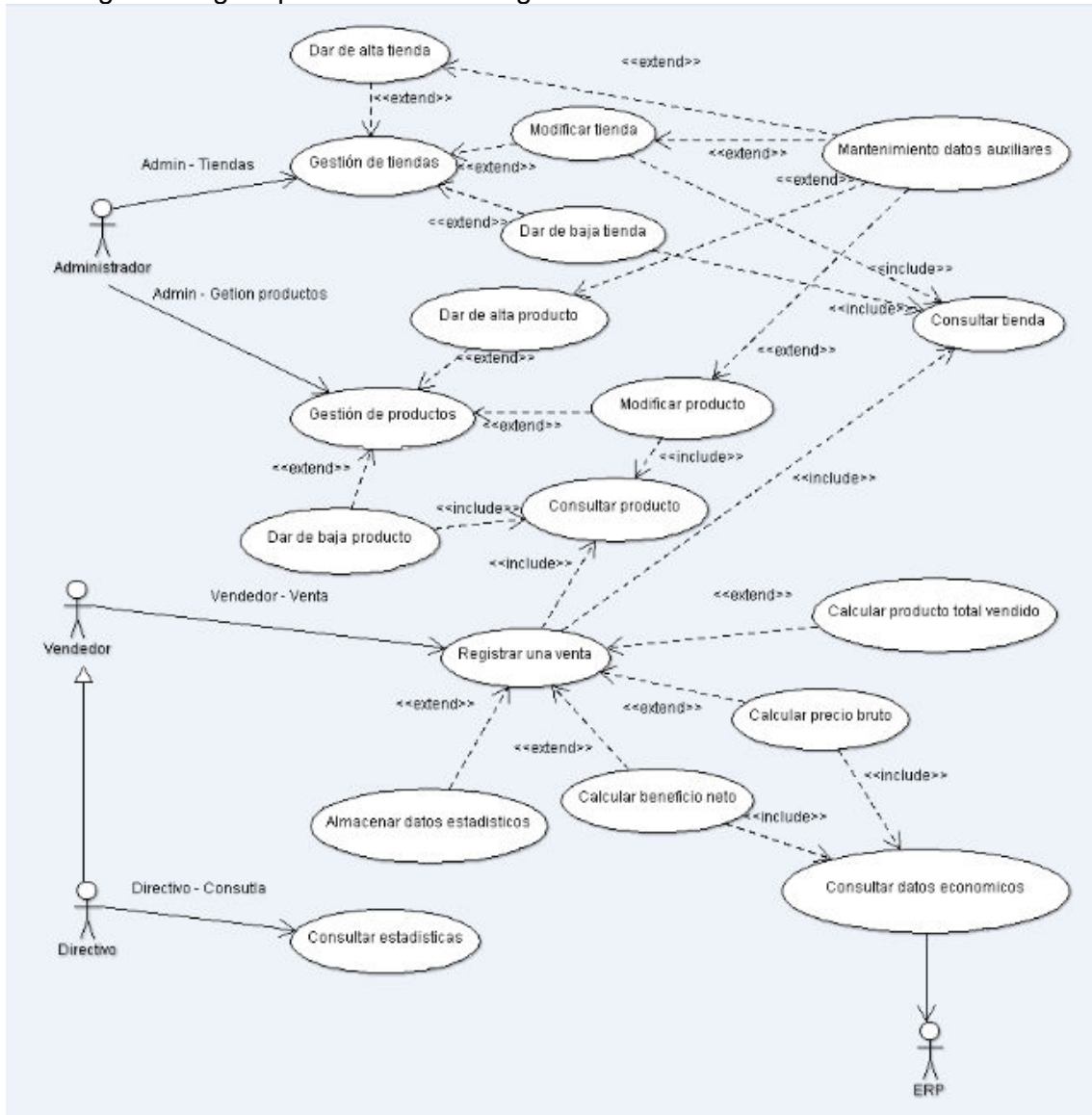


Figura 13: Diagrama de casos de uso.

A continuación se especifican los diferentes casos de uso.

Caso de uso: Gestión de tiendas

Resumen de la funcionalidad: Mediante este caso de uso el administrador podrá acceder a las diferentes acciones para el mantenimiento de los datos relativos a las tiendas.

Precondición: El administrador ha de estar registrado en el sistema.

Postcondición: Ninguna.

Caso de uso: Dar de alta tienda

Resumen de la funcionalidad: Se informan los datos relativos a una tienda nueva y éstos se graban en la BD.

Precondición: El administrador ha elegido la opción para dar de alta una tienda.

Postcondición: La nueva tienda queda registrada en el sistema.

Caso de uso: Dar de baja tienda

Resumen de la funcionalidad: Se elimina en la Base de Datos una tienda determinada y la información relativa a las ventas de esa tienda.

Precondición: El administrador ha elegido la opción para dar de baja una tienda. La tienda existe en el sistema.

Postcondición: La tienda no existe en el sistema.

Caso de uso: Modificar tienda

Resumen de la funcionalidad: Se modifican los datos referentes a una tienda determinada.

Precondición: El administrador ha elegido la opción para modificar una tienda. La tienda existe en el sistema.

Postcondición: Los datos de la tienda quedan modificados en la BD.

Caso de uso: Consultar tienda

Resumen de la funcionalidad: Busca una tienda a través de su identificador en la BD y devuelve los datos de dicha tienda.

Precondición: Se ha introducido un identificador de tienda en el campo correspondiente.

Postcondición: Ninguna.

Caso de uso: Mantenimiento datos auxiliares

Resumen de la funcionalidad: Se consultan y cargan los datos auxiliares para el mantenimiento de las tiendas y del catálogo de productos.

Precondición: Se ha solicitado un alta de tienda o de producto a través del caso de uso correspondiente.

Postcondición: Los nuevos datos quedan registrados en la BD.

Caso de uso: Gestión de productos

Resumen de la funcionalidad: Mediante este caso de uso el administrador podrá acceder a las diferentes acciones de mantenimiento de los datos relativos a los productos.

Precondición: El administrador ha de estar registrado en el sistema.

Postcondición: Ninguna.

Caso de uso: Dar de alta producto

Resumen de la funcionalidad: Se cargan los datos relativos a un producto nuevo en la base de datos.

Precondición: El administrador ha elegido la opción para dar de alta un producto.

Postcondición: El nuevo producto queda registrado en el sistema.

Caso de uso: Dar de baja producto

Resumen de la funcionalidad: Se elimina de la Base de Datos un producto determinado y la información relativa a las ventas de ese producto.

Precondición: El administrador ha elegido la opción para dar de baja un producto. El producto existe en el sistema.

Postcondición: El producto no existe en el sistema.

Caso de uso: Modificar producto

Resumen de la funcionalidad: Se modifican los datos referentes a un producto determinado.

Precondición: El administrador ha elegido la opción para modificar un producto. El producto existe en el sistema.

Postcondición: Los datos del producto quedan modificados en la BD.

Caso de uso: Consultar producto

Resumen de la funcionalidad: Busca un producto a través de su identificador en la BD y devuelve los datos de dicho producto.

Precondición: Se ha introducido un identificador de producto en el campo correspondiente.

Postcondición: Ninguna.

Caso de uso: Registrar venta

Resumen de la funcionalidad: Se registran los datos referentes a una venta en la BD.

Precondición: El vendedor o directivo está registrado en el sistema. La tienda existe en el sistema. El producto existe en el sistema.

Postcondición: La venta queda registrada en la BD.

Caso de uso: Calcular producto total vendido

Resumen de la funcionalidad: Se calculan las unidades de producto con el mismo código EAN13 vendidas hasta el momento en el mismo día y hora incluyendo la solicitud de registro que se ha efectuado.

Precondición: Se ha solicitado el registro de una venta.

Postcondición: Ninguna.

Caso de uso: Calcular precio bruto

Resumen de la funcionalidad: Se calcula el precio bruto total de los productos con el mismo código EAN13 vendidos hasta el momento en el mismo día y hora incluyendo la solicitud de registro que se ha efectuado.

Precondición: Se ha solicitado el registro de una venta.

Postcondición: Ninguna.

Caso de uso: Calcular beneficio neto

Resumen de la funcionalidad: Se calcula el beneficio neto total de los productos con el mismo código EAN13 vendidos hasta el momento en el mismo día y hora incluyendo la solicitud de registro que se ha efectuado.

Precondición: Se ha solicitado el registro de una venta.

Postcondición: Ninguna.

Caso de uso: Almacenar datos estadísticos

Resumen de la funcionalidad: Registrar en la BD determinados ratios estadísticos para futura consulta directa (estos ratios se verán con detalle en el punto 3.3, apartado D, pág. 29).

Precondición: Se ha registrado una venta.

Postcondición: Los ratios estadísticos quedan registrados en la BD.

Caso de uso: Consultar estadísticas

Resumen de la funcionalidad: Mostrar los datos estadísticos de las ventas.

Precondición: El directivo ha de estar registrado en el sistema.

Postcondición: Ninguna.

Caso de uso: Consultar datos económicos

Resumen de la funcionalidad: Obtener del ERP los datos económicos requeridos para el registro de una venta, es decir, el precio bruto y el beneficio neto.

Precondición: Ha de existir conexión del sistema con el ERP.

Postcondición: Ninguna.

3.3 Requerimientos funcionales

Como indicaba en el primer apartado de este capítulo, los requisitos funcionales describen lo que debe hacer el software. Tras el análisis del texto del enunciado y las diferentes consultas y conversaciones con el cliente se han detectado cuatro requisitos funcionales que se desarrollan a continuación.

Cada uno de los cuatro requisitos se desarrolla identificando los datos que debe contener la Base de Datos para dar respuesta a las necesidades del cliente y al final se enumera una lista con las acciones que el sistema debe poder ejecutar sobre dichos datos o conjunto de datos. Para cada dato se da una breve explicación así como las restricciones que lleva asociadas. Aquellos datos que están sombreados en azul son los que constan en la especificación del enunciado. Los datos que están sombreados en gris son propuestas de mejora. Para cada propuesta de mejora se explica por qué se ha propuesto y si el cliente la ha aceptado o rechazado.

A. Almacenar información diversa sobre las tiendas de la cadena de ropa.

La base de datos debe permitir almacenar información relativa a las tiendas que conforman la cadena. Dicha información ha de ser útil para identificar una tienda concreta dentro del grupo de tiendas. Además debe proporcionar datos básicos de su ubicación y datos de contacto.

A continuación se muestra una lista con los datos que debe contener la BD para dar respuesta a este requisito que se corresponde con el apartado [R1] del enunciado.

A.1 Identificador de la tienda

Descripción: Este identificador debe permitir identificar inequívocamente una tienda dentro del conjunto de tiendas que conforman la cadena. Ha de funcionar como una clave primaria, es decir, no se puede repetir para dos tiendas.

Restricciones: Lo debe proporcionar el sistema al dar de alta una tienda y es obligatorio.

A.2 Ciudad de la tienda

Descripción: Ciudad donde se ubica la tienda.

Restricciones: Dadas las características inherentes a una tienda física, ésta no puede cambiar de ciudad. Sin embargo si podría cambiar de ciudad una tienda de venta online. Es obligatorio.

A.3 Región de la tienda

Descripción: La región a la que pertenece la ciudad donde se ubica la tienda.

Restricciones: Dadas las características inherentes a una tienda física, ésta no puede cambiar de región. Sin embargo si podría cambiar de región una tienda de venta online. Es obligatorio.

A.4 Correo electrónico del gerente de la tienda

Descripción: Dirección de correo electrónico de la persona que ostenta el cargo de gerente de la tienda.

Restricciones: Cada tienda tiene un gerente pero el puesto puede estar vacante*.

A.5 Número de trabajadores de la tienda

Descripción: Recoge el número de empleados que trabajan en la tienda en un momento determinado. El número de empleados puede variar con el tiempo*.

Restricciones: Es un dato obligatorio.

A.6 Identificador de régimen de explotación.

Descripción: Debe mostrar si la tienda es propiedad al 100% de la cadena o si la tienda está explotada en régimen de franquicia.

Restricciones: debe ser un dato booleano y obligatorio.

A.7 Identificador del tipo de tienda.

Descripción: Debe mostrar si la tienda es una tienda de venta virtual o bien es una tienda física con un emplazamiento concreto para la exposición y venta al público.

Restricciones: debe ser un dato booleano y obligatorio.

A.8 Identificador País donde se ubica la tienda.

Descripción: El país al que pertenece la región donde se ubica la tienda.

Motivación de propuesta de inclusión: En el contexto actual de internacionalización de mercados es conveniente dejar abierta la opción de indicar el país de la tienda de cara a un posible crecimiento. Aceptado por el cliente el 23.10.2015*.

Restricciones: Dadas las características inherentes a una tienda física, ésta no puede cambiar de País. Sin embargo si podría cambiar de País una tienda de venta online.

A.9 Dirección postal.

Descripción: La dirección postal de la tienda. Debe incluir calle, número y código postal.

Motivación de propuesta de inclusión: Dado que en la base de datos están recogidos todos los datos relativos a las tiendas se puede complementar fácilmente con la dirección postal para los establecimientos físicos de cara a futuras estadísticas que apoyen acciones comerciales así como para poder analizar las distancias entre las diferentes tiendas. Aceptado por el cliente el 23.10.2015*.

*Ver Anexo I

Restricciones: Dadas las características inherentes a una tienda física, ésta no puede cambiar de dirección postal. Sin embargo si podría cambiar de dirección postal una tienda de venta online.

A.8 Fecha de apertura.

Descripción: La fecha en que la tienda se ha abierto al público.

Motivación de propuesta de inclusión: Puede ser interesante para futuras estadísticas que contemplen evolución de aperturas de tiendas de la cadena en el tiempo. Rechazado por el cliente el 23.10.2015 ya que no quieren conservar datos de tiendas cerradas*.

Restricciones: Este dato ha de ser de formato dd/mm/aaaa y no puede ser eliminado ni alterado. Es un dato obligatorio.

A.9 Fecha de cierre.

Descripción: La fecha en que la tienda se ha cerrado al público.

Motivación de propuesta de inclusión: Dado que una tienda, por razones de integridad, no debería ser eliminada una vez tiene ventas registradas, es conveniente poder registrar el periodo a partir del cual no puede registrar más ventas al estar cerrada para la venta al público. Rechazado por el cliente el 23.10.2015 ya que no quieren conservar datos de tiendas cerradas*.

Restricciones: Este dato ha de ser de formato dd/mm/aaaa.

En relación con este primer requisito el sistema debe proporcionar, considerando las restricciones expuestas anteriormente, las siguientes funcionalidades:

- Dar de alta una tienda.
- Dar de baja una tienda. Se debe permitir aunque tenga ventas asociadas ya que en caso de cierre de la tienda no hay que conservar los datos de la misma*. Por tanto, al borrar una tienda del sistema, se tienen que eliminar todos los datos relativos a las ventas de esa tienda y actualizar las estadísticas.
- Modificar los datos de una tienda.
- No es necesario implementar los procedimientos de alta, baja y modificación para los datos maestros.

B. Almacenar el número de ventas realizadas diariamente de cada uno de los productos disponibles en el catálogo de productos de la cadena.

También la BD debe permitir almacenar las ventas que se realizan en el conjunto de tiendas de la cadena. Para saber los productos que vende la cadena existe un catálogo que se ha de poder gestionar con esta BD y que es común para todas las tiendas.

*Ver Anexo I

Hay que destacar que la gestión económica del inventario se realiza en el ERP de la cadena de tiendas. Por tanto no debe esta BD guardar el histórico de precios, ni gestionar el IVA de las ventas. Tampoco se establecerán controles de stock. Por tanto los productos que estén en el catálogo es que están disponibles para la venta.

A continuación se muestra una lista con los datos básicos que debe contener la BD para dar respuesta a este requisito que se corresponde con los apartados [R2] y [R3] del enunciado.

Por lo que se refiere al catálogo:

B.1 Identificador EAN13 del producto.

Descripción: Debe contener el código EAN13 que identifica el producto en cuestión. El código que se introduzca se considera correcto por lo que no es necesario comprobar su validez*.

Restricciones: Debe contener 13 dígitos, no se puede modificar y es obligatorio.

B.2 Descripción del producto.

Descripción: Se debe describir el producto en sus principales elementos como tipo de tela, color, etc.

Restricciones: Se debe permitir una longitud de texto considerablemente amplia pues la descripción puede ser larga. Es un dato obligatorio.

B.3 Fecha de incorporación al catálogo.

Descripción: La fecha en que el producto entra a formar parte del catálogo y representa el momento inicial en el que el producto queda disponible para la venta.

Restricciones: Debe tener el formato dd/mm/aaaa. Es un dato obligatorio.

B.4 Tipo de producto.

Descripción: Categoría a la que puede pertenecer el producto.

Motivación de propuesta de inclusión: En un futuro puede ser interesante observar estadísticas de ventas sobre tipos de productos para poder mejorar las acciones comerciales. Aceptado por el cliente el 23.10.2015*.

Restricciones: Debe proporcionar una lista de categorías corta pero ampliable.

B.5 Género del producto.

Descripción: Categoría a la que puede pertenecer el producto en función del consumidor final.

Motivación de propuesta de inclusión: En un futuro puede ser interesante observar estadísticas de ventas sobre tipos de productos para poder mejorar las acciones comerciales. Aceptado por el cliente el 23.10.2015*.

*Ver Anexo I

Restricciones: Debe proporcionar una lista de categorías corta y cerrada con los siguientes elementos: hombre, mujer, niño, niña, unisex.

B.6 Fecha de baja del catálogo.

Descripción: Fecha en que el producto deja de pertenecer al catálogo.

Motivación de propuesta de inclusión: Permitirá en un futuro saber si una prenda está en producción o no. Al existir ventas de un producto, éste no se debería eliminar de la BD y por tanto siempre aparecería disponible para la venta. Rechazado por el cliente el 23.10.2015 ya que no quieren conservar datos de los productos dados de baja*.

Restricciones: Debe tener el formato dd/mm/aaaa.

Por lo que se refiere a la venta:

B.9 Identificador de la tienda.

Descripción: Tienda en la que se realiza la venta.

Restricciones: El identificador de tienda ha de existir en el sistema. Este dato debe formar parte de la identificación única de una venta.

B.10 Identificador EAN13 del producto.

Descripción: Código EAN13 que identifica el producto que se ha vendido.

Restricciones: El código ha de existir en el sistema. Este dato debe formar parte de la identificación de una venta.

B.11 Fecha de la venta.

Descripción: Fecha en que se realiza la venta.

Restricciones: Tiene que tener el formato dd/mm/aaaa. Este dato debe formar parte de la identificación de una venta.

B.12 Hora de la venta.

Descripción: Hora en que se realiza la venta.

Restricciones: Tiene que ser en formato 24 horas y sin especificar el minuto de la venta. Sólo puede admitir valores entre 0 y 23. Este dato debe formar parte de la identificación de una venta.

B.13 Cantidad total de producto vendida aquel día y hora.

Descripción: Indica la cantidad total de unidades vendidas con el mismo código EAN13 en ese mismo día y hora en todas las tiendas de la cadena.

Restricciones: No puede tener valores negativos. Puede tener valor 0 para el caso de una única venta y una devolución.

*Ver Anexo I

B.14 Precio bruto total de los productos vendidos aquel día y hora.

Descripción: Indica el importe bruto total de todas las ventas realizadas para ese código EAN13 en ese día y hora en todas las tiendas de la cadena.

Restricciones: No puede tener valores negativos. Puede tener valor 0 para el caso de una única venta y una devolución. El precio de la venta viene determinado por el ERP externo.

B.15 Beneficio neto total de los productos vendidos aquel día y hora.

Descripción: Indica el beneficio neto total de todas las ventas realizadas para ese código EAN13 en ese día y hora en todas las tiendas de la cadena.

Restricciones: Se debe calcular como diferencia entre precio de venta y coste del producto vendido. El margen de la venta viene dado por el ERP externo.

B.16 Unidades vendidas.

Descripción: Número de unidades que se venden del producto con ese EAN13.

Motivación de propuesta de inclusión: En una venta puede venderse más de una unidad, aunque no es frecuente si puede ocurrir en prendas pequeñas. Cabe especificar por tanto el número de unidades que se venden en cada venta.

Restricciones: Ha de ser un número entero.

B.19 Empleado que hace la venta.

Descripción: Código identificador del empleado que hace la venta.

Motivación de propuesta de inclusión: Habiendo un registro de ventas es muy útil saber quién ha hecho la venta. De ese modo se permite una gestión eficiente de retribuciones variables en función de las ventas que suelen tener este tipo de empleados. Rechazado por el cliente el 23.10.2015 ya que no quieren hacer gestión de retribuciones variables en función de ventas*.

Restricciones: El código debe existir fuera del sistema. Es un dato obligatorio.

En relación con este requisito el sistema debe proporcionar, considerando las restricciones expuestas anteriormente, las siguientes funcionalidades:

- Dar de alta una prenda en el catálogo.
- Dar de baja una prenda en el catálogo. Por tanto, al borrar un producto del sistema se tienen que eliminar todos los datos relativos a las ventas de ese producto y actualizar las estadísticas.
- Modificar una prenda en el catálogo.
- Dar de alta una venta.

El sistema no debería permitir dar de baja ni modificar una venta. Si se permitiera podría dar lugar a manipulaciones malintencionadas de las estadísticas. Por tanto es más conveniente que si ha habido algún error este se subsane agregando una venta indicando en el dato A.16 una cifra negativa.

*Ver Anexo I

C. Proporcionar, en tiempo de ejecución de la consulta, determinada información estadística de ventas.

La base de datos debe permitir acceder a los siguientes datos económicos referidos a las ventas. Esta información debe ser calculada en el momento de realizar la consulta y no es necesario almacenarla para futuras consultas. Se corresponde con el apartado [R4] del enunciado.

C.1 Estadística de ventas por tienda para un mes y año concretos.

Descripción: Dados un año y un mes debe proporcionar el listado de todas las tiendas de la cadena incluyendo para cada tienda:

- Número de unidades total de todos los productos vendidos en el periodo indicado como parámetro en la consulta, es decir, el total de unidades vendidas en esa tienda.
- Número de productos diferentes vendidos en el periodo indicado como parámetro en la consulta, es decir, número de códigos EAN13 diferentes vendidos en ese mes.
- Beneficio neto total obtenido en el periodo indicado como parámetro en la consulta.
- Porcentaje de beneficio que aporta cada tienda en relación con el total de beneficios obtenidos por toda la cadena en el periodo indicado como parámetro en la consulta.
- El beneficio neto dividido por el número de empleados de la tienda ese mes.

Restricciones: El listado debe mostrarse ordenado de forma descendente según el beneficio neto total. Debe ser accesible sólo por los directivos.

C.2 Estadística de ventas por producto para un mes y año concretos.

Descripción: Dados un año y un mes debe proporcionar el listado de todos los productos del catálogo incluyendo para cada producto:

- Identificador EAN13.
- Nombre del producto.
- Número de unidades vendidas del producto.
- Beneficio neto que ha generado el producto.
- Identificador de la tienda que ha vendido más unidades.
- Número de unidades que ha vendido la tienda del punto anterior.

Restricciones: El listado debe mostrarse ordenado de forma descendente según el beneficio generado por el producto. Debe ser accesible sólo por los directivos.

C.3 Estadística de ventas por días para un mes y año concretos.

Descripción: Dados un año y un mes debe proporcionar el listado de todos los días del mes incluyendo para cada día:

- Beneficio total neto obtenido aquel día para toda la cadena.

- Identificador EAN13 del producto más vendido así como el número de unidades vendidas de dicho producto.
- Identificador de la tienda de la cadena que más beneficio neto ha obtenido.
- Valor en Euros del beneficio atribuible a la tienda del apartado anterior.

Restricciones: El listado debe mostrarse por días, del primer al último día del mes. Debe ser accesible sólo por los directivos.

C.4 Estadística de ventas por empleado para un mes y año concretos.

Descripción: Dados un año y un mes debe proporcionar el listado de todos los empleados incluyendo para cada día:

- Identificador del empleado.
- Número de unidades vendidas.
- Número de productos diferentes vendidos.
- Beneficio neto que ha generado ese empleado.

Restricciones: El listado debe mostrarse ordenado de forma descendente según el beneficio generado por el empleado. Debe ser accesible sólo por los directivos. Rechazado por el cliente el 23.10.2015 ya que no quieren hacer gestión de retribuciones variables en función de ventas*.

D. Proporcionar información consultable en tiempo real por la dirección de la cadena que permita obtener decisiones sobre el negocio.

El sistema ha de permitir acceder a los siguientes listados de ratios económicos en un tiempo constante 1. Es decir, deben ser accesibles mediante una consulta sencilla sobre un registro en una tabla y no deben ser calculados en el momento de hacer la consulta. Se corresponde con el apartado [R5] del enunciado.

D.1 Estadística de ventas de un año concreto.

Descripción: Dado un año debe proporcionar los siguientes datos:

- Beneficio total neto de toda la cadena.
- Identificador de la tienda que más beneficios netos ha conseguido
- La cifra de beneficio neto, expresada en Euros, de la tienda anterior.
- Identificador EAN13 del producto más vendido.
- La cantidad total de unidades vendidas del producto del apartado anterior.
- La hora del día en la que se más productos se han vendido.
- La cifra total de unidades vendidas en la hora del apartado anterior.
- La hora del día donde menos productos se han vendido.
- La cifra total de unidades vendidas en la hora del apartado anterior.
- El día del mes en el que se han realizado más venta.
- El número de unidades total vendidas en el día del apartado anterior.
- El día del mes en el que se han realizado menos ventas.

*Ver Anexo I

- El número de unidades total vendidas en el día del apartado anterior.
- Ciudad donde más beneficios netos se han obtenido.
- Beneficio de la ciudad del apartado anterior.
- Porcentaje de beneficios obtenidos por tiendas virtuales respecto del total de beneficios de la cadena.

Restricciones: Debe ser accesible sólo por los directivos. Las respuestas han de estar siempre actualizadas con la última información de la BD por lo que no se pueden utilizar procesos que se ejecuten por la noche para alimentar las estadísticas. No se pueden emplear vistas calculadas ni funciones de agregación.

D.2 Estadística de ventas totales.

Descripción: En el momento de ejecutar la consulta se debe obtener un listado con la misma información del punto D.1 pero para todas las ventas de la cadena.

Restricciones: Debe ser accesible sólo por los directivos. Las respuestas han de estar siempre actualizadas con la última información de la BD por lo que no se pueden utilizar procesos que se ejecuten por la noche para alimentar las estadísticas. No se pueden emplear vistas calculadas ni funciones de agregación.

3.4 Requerimientos no funcionales

Como ya se ha indicado, los requerimientos no funcionales son aquellos comportamientos del sistema que no se asocian a ningún caso de uso concreto. Se podría decir que refieren a las propiedades del sistema, no a las funcionalidades.

Tras la lectura del enunciado se han puesto de relieve los siguientes requisitos no funcionales.

- Como mínimo tiene que haber un parámetro de salida llamado RSP, de tipo *String*, que indicará si la ejecución ha terminado con éxito (valor "OK") o si no se ha podido realizar (valor "ERROR+TIPO DE ERROR").
- Ha de disponer de tratamiento de excepciones.
- Se han de almacenar todas las llamadas a los procedimientos en una tabla de log donde se han de almacenar el procedimiento ejecutado y los parámetros de entrada y de salida.
- Los procedimientos han de estar especificados para que no haga falta mirar el código por los programadores de la capa superior. Como mínimo la documentación tiene que contener la descripción de lo que hace y el tipo y valores posibles de los parámetros de entrada y de salida incluyendo los códigos de error que puede devolver y su significado.
- El código que se genere ha de tener comentarios que expliquen su comportamiento interno.

4. Diseño de la base de datos

En este apartado se va a tratar el diseño de la base de datos considerando el análisis de requisitos del apartado anterior. El diseño se hará en tres fases, diseño conceptual, diseño lógico y diseño físico.

4.1 Diseño conceptual

Tras el análisis realizado de los requisitos expuestos por el cliente y detallados en este documento se puede definir la estructura que va a tener la base de datos. Como se apuntó en el apartado 2.4, este diseño se va a realizar mediante un modelo E/R lo que nos permite no tener en consideración la tecnología que se va a emplear.

En la siguiente figura se muestra el modelo conceptual de la base de datos.

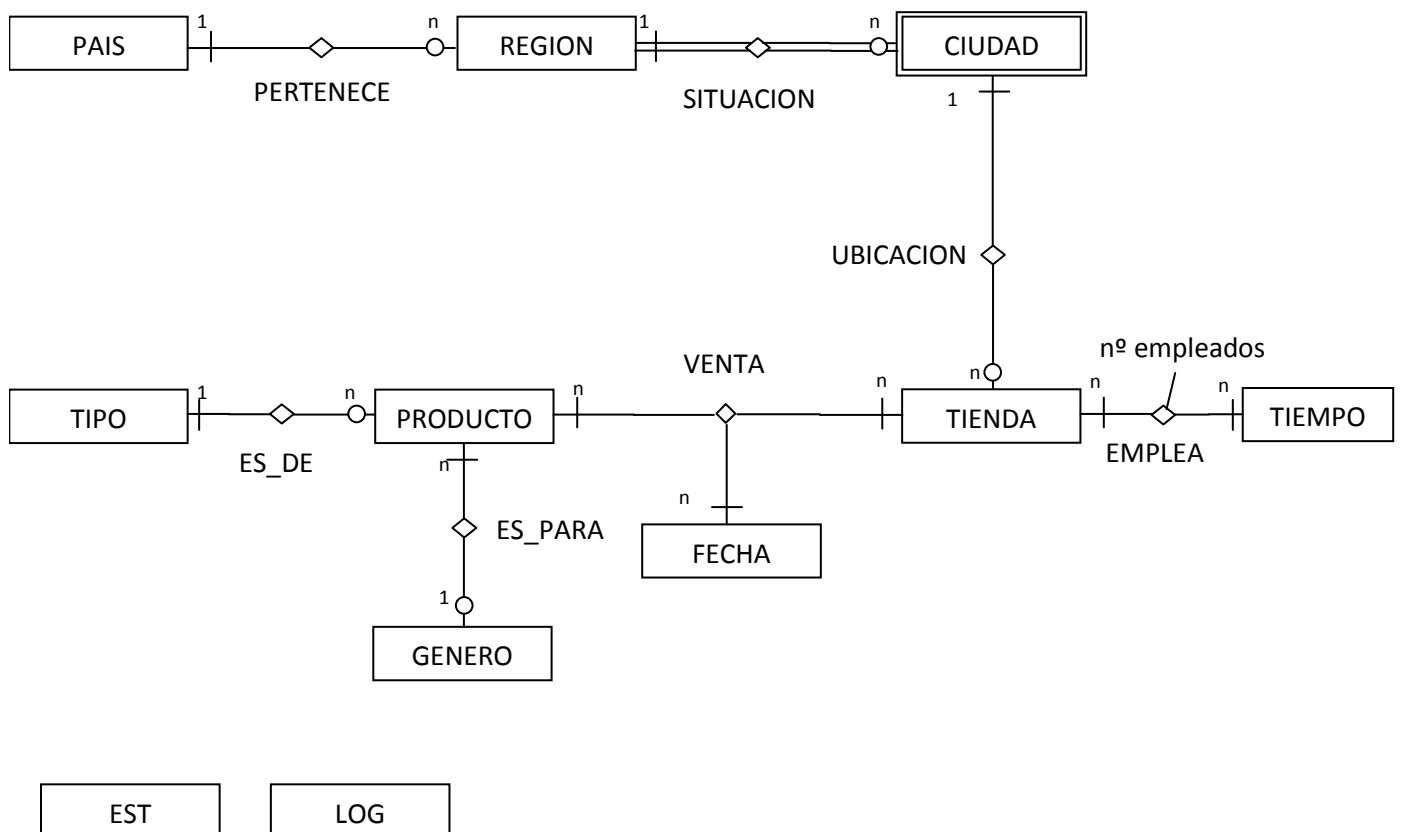


Figura 14: Diagrama E/R.

PAIS: nombre país.

REGION: nombre región.

CIUDAD (entidad débil: nombre_ciudad la identifica parcialmente, se identifica completamente con la región de situación): nombre_ciudad.

TIENDA: identificador, nombre_ciudad, correo_gerente, régimen_explotación, tipo_tienda, dirección_postal.

TIEMPO: año, mes.

PRODUCTO: EAN13, descripción, fecha_incorporación, tipo, genero.

GENERO: nombre.

TIPO: nombre.

FECHA: fecha, hora.

EST: id, beneficio_neto, identificador, cifra_beneficio, EAN13, cantidad_unidades_producto, hora_día_max, cantidad_unidades_hora_max, hora_día_menos, cantidad_unidades_hora_menos, día_mes_max, unidades_día_mes, día_mes_menos, unidades_día_mes_menos, ciudad_más_beneficios, beneficio_ciudad, porcentaje_beneficios_virtuales.

LOG: id, procedimiento, fecha, parámetros_entrada, parámetros_salida.

A continuación se exponen los argumentos que justifican el diseño.

- La entidad PAIS es un dato auxiliar de la BD. Es obligatoria en la relación con REGION por que no puede haber una región que no pertenezca a un país.
- La entidad CIUDAD se ha considerado entidad débil ya que puede haber ciudades con el mismo nombre en países diferentes. Por tanto para identificarla totalmente es necesario saber la región en la que se encuentra. Esa entidad se podría sustituir en el modelo por una entidad asociativa pero dado que el esquema es relativamente sencillo no se considera necesario.
- La VENTA es una relación ternaria entre PRODUCTO, TIENDA y FECHA ya que por exigencia del cliente tiene que estar identificada con esos tres datos. A parte, dicha relación tiene una serie de atributos que no se muestran en el gráfico por simplicidad y que son “cantidad total de producto vendido”, “precio bruto total de los productos vendidos”, “beneficio neto total de los productos vendidos” y “unidades vendidas”. Para más detalle se puede consultar los puntos 3.3 B13, 3.3 B14, 3.3 B15 y 3.3 B16.
- Debido a que el cliente ha aceptado el catalogar los productos en diferentes tipos se refleja en el modelo la posibilidad de relacionarlo con distintos tipos o diferentes géneros según el usuario final.
- El cliente también ha comentado que el número de empleados varía según el mes. Es necesario por tanto establecer una relación con la entidad TIEMPO para poder almacenar el número de empleados por tienda y mes.

4.2 Diseño lógico

En este punto se considera el tipo de Base de Datos que se va a implementar. Por exigencias de enunciado se va a desarrollar una Base de Datos relacional y por tanto se va a transformar el diagrama E/R de la página 31 en una estructura de datos según el modelo relacional.

En el caso que estamos tratando tendremos que observar los siguientes aspectos para la transformación:

- Cada entidad del modelo E/R dará lugar a una relación y los atributos serán los mismos.
- Para transformar las interrelaciones 1:n se pondrá en la relación del lado n una clave foránea que referencie a la relación del lado 1.
- Cada interrelación n:n darán lugar a una relación cuya clave primaria viene formada por los atributos de las claves primarias de las entidades que se interrelacionan.
- Una interrelación ternaria n:n:n será una nueva relación cuya clave primaria tiene todos los atributos que forman las claves primarias de las entidades relacionadas.
- En caso de las entidades débiles la clave foránea que hace referencia a la relación que la identifica debe formar parte de la clave primaria de la entidad débil.

Según lo anterior, las relaciones que habría que crear son las siguientes:

PAIS (nombre_país)

REGION (nombre_región, nombre_país)
donde {nombre_país} referencia a PAIS

CIUDAD (nombre_ciudad, nombre_región)
donde {nombre_región} referencia a REGION

TIENDA (identificador, nombre_ciudad, correo_gerente, régimen_explotación, tipo_tienda, dirección_postal)
donde {nombre ciudad} referencia a CIUDAD

TIEMPO (año, mes)

PRODUCTO (EAN13, descripción, fecha_incorporación, tipo, género)
donde {tipo} referencia a TIPO, {genero} referencia a GENERO

GENERO (nombre)

TIPO (nombre)

FECHA (fecha, hora)

NUMERO_EMPLEADOS (identificador, año, mes, número_empleados)
donde {identificador} referencia a TIENDA, {año, mes} referencia a TIEMPO

VENTA (identificador, EAN13, fecha, hora, cantidad_total, precio_bruto_total, beneficio_neto_total, unidades_vendidas)
donde {identificador} hace referencia a TIENDA, {EAN13} referencia a PRODUCTO, {fecha, hora} referencia a FECHA

EST (id, beneficio_neto, identificador, cifra_beneficio, EAN13, cantidad_unidades_producto, hora_día_max, cantidad_unidades_hora_max, hora_día_menos, cantidad_unidades_hora_menos, día_mes_max, unidades_día_mes, día_mes_menos, unidades_día_mes_menos, ciudad_más_beneficios, beneficio_ciudad, porcentaje_beneficios_virtuales).

LOG (Id, procedimiento, fecha, parámetros_entrada, parámetros_salida).

4.3 Diseño físico

El diseño físico es la especificación de cómo se van a guardar los datos en la BD. Por ello es importante conocer en detalle las funcionalidades que ofrece el SGBD que se va a usar de modo que se pueda hacer una transformación correcta del diseño lógico.

Generalmente el diseño físico consta de 4 etapas, cada una con sus sub-fases:

- Traducir el esquema lógico para el SGBD específico.
 1. Diseñar las relaciones.
 2. Diseñar las reglas de la BD.
- Diseñar la representación física.
 1. Analizar las transacciones.
 2. Organizar los ficheros.
 3. Escoger los índices secundarios.
 4. Estimar la necesidad de espacio en disco.
- Diseñar los mecanismos de seguridad.
 1. Diseñar las vistas de los usuarios.
 2. Fijar las reglas de acceso.
- Monitorizar y afinar el sistema.

Tal y como se fijó en la fase de calificación (capítulo 2), sólo trataré el primer punto del diseño físico abordando el diseño de las relaciones y las reglas de la BD. Se entiende que Oracle DB XE 11g. es un herramienta lo suficientemente potente como para dar respuesta a cualquier necesidad de organización de ficheros y acceso a datos que pudiera ser necesaria para esta aplicación.

4.3.1 Características Oracle Express 11g

Oracle Database 11g Express Edition es un pequeño motor de Base de Datos basado en el código de Oracle Database 11g Release 2. Puede ser instalado en cualquier

máquina con 1 CPU, 1 GB de memoria RAM así como 4GB de espacio en disco para la instalación básica. Está soportado tanto por Linux como por Windows.

A continuación se muestran los requisitos mínimos para Windows 32-bit. Información mucho más ampliada puede ser obtenida del sitio oficial de Oracle:

http://docs.oracle.com/cd/E17781_01/install.112/e18803/toc.htm#XEINW102

Requirement	Value
System architecture	Intel (x86), AMD64, and Intel EM64T
Operating system	One of the following 32-bit Microsoft Windows operating systems: Microsoft Windows Server 2008 - Standard, Enterprise, Datacenter, Web, and Foundation Editions. The Server Core option is not supported. Microsoft Windows 7 - Professional, Enterprise, and Ultimate Editions Windows 8 - Pro and Enterprise Editions
Network protocol	The following protocols are supported: IPC Named Pipes SDP TCP/IP TCP/IP with SSL
Disk space	1.5 gigabytes minimum
RAM	256 megabytes minimum, 512 megabytes recommended for Oracle Database XE. The operating system itself may have a higher minimum requirement.
Microsoft Windows Installer (MSI)	MSI version 2.0 or later

Figura 15: Oracle Database XE Requerimientos para Microsoft Windows 32-bit

Respecto a las características de diseño de tablas y reglas de BD se puede apuntar de modo abreviado lo siguiente:

- Permite definir claves primarias y foráneas.
- Permite definir todos los tipos de datos requeridos en las relaciones definidas en durante el diseño lógico **salvo el tipo booleano**.
- Se pueden definir dominios para los atributos.
- Se pueden definir valores nulos para los atributos.
- Permite la creación de paquetes.
- Permite la creación de Triggers para poder implementar determinadas reglas de negocio.

Es importante remarcar que no se puede definir el tipo booleano como tipo de dato para una columna de una tabla (si se puede en funciones y procedimientos). Uno de los requisitos del enunciado de la práctica exige que se definan los campos de franquicia y de tienda virtual como campos booleanos. En el apartado 4.3.3 se detallan las posibles soluciones y la solución escogida.

4.3.2 Nomenclatura

Para la elaboración del código se van a seguir las siguientes convenciones que están en línea con aquellas establecidas para Bases de Datos de Oracle (sólo se definen aquellas que serán utilizadas en la fase de programación para no hacer este apartado excesivamente extenso):

- Se seguirá la convención underscore.
- Si un nombre es demasiado largo se acortará eliminando todas las vocales menos la primera de cada una de las palabras que lo componen. Si aún así no se acorta lo suficiente se escogerán abreviaturas que se usen de forma general o que hagan clara referencia a su significado.
- Los nombres de las tablas serán en plural, mayúsculas y auto descriptivos. No llevarán ningún sufijo ni prefijo. No pueden tener más de 30 caracteres y sólo pueden contener caracteres ASCII y el símbolo '_'.
- Los nombres de las columnas serán en singular, mayúsculas y auto descriptivos. No llevarán ningún sufijo ni prefijo para referenciar a su tabla. No pueden tener más de 30 caracteres sólo pueden contener caracteres ASCII y el símbolo '_'.
- Los nombres de las columnas que representan lo mismo en diferentes tablas se llamarán de la misma manera.
- Los objetos que hagan referencia a una tabla irán precedidos del nombre de la tabla y de un sufijo que los identifique. Así por ejemplo una clave primaria se nombrará como <TABLA>_PK, una clave foránea como <TABLA>_FK, un trigger como <TABLA>_TR, etc. Si hay varios objetos con el mismo nombre se numerarán a continuación del sufijo.
- Los paquetes de la Base de Datos irán precedidos por el prefijo PKG_.
- Los tipos de dato creados por el usuario irán precedidos por el prefijo T_ y se nombrarán en mayúsculas con caracteres ASCII.
- Los tipos de dato que sean tabla de otro tipo de dato se nombrarán como T_TABLA_<TIPO DE DATO>.
- Las secuencias irán precedidas por el prefijo SEQ_ y se nombrarán en mayúsculas con caracteres ASCII. Si son usadas exclusivamente por una tabla se pondrá el nombre de la tabla a continuación del prefijo.
- Las funciones o procedimientos no se identifican como tales mediante sufijo o prefijo.
- Los parámetros de entrada tanto de funciones como de procedimientos se preceden del prefijo e_.

4.3.3 Diseño de relaciones

Para definir las relaciones se usará el lenguaje DDL (Lenguaje de Definición de Datos) del SGBD. Por tanto, en este caso usaré PL/SQL. Hay que definir para cada relación el nombre de la tabla, las columnas (atributos) y su dominio, valor por defecto y si admite nulos.

En particular hay que observar los siguientes hechos que influyen en la construcción final de la aplicación y que no quedan reflejados en el diseño lógico:

- Para facilitar las consultas que se hacen sobre la BD se recomienda incluir dos columnas más en la tabla de ventas indicando el precio de venta bruto unitario y el beneficio neto unitario de cada transacción. De ese modo se van a poder realizar consultas mucho más ágiles.
- Como se ha apuntado en el punto 4.3.1 la BD no permite el uso de tipos booleanos para las columnas. Hay dos soluciones posibles generalmente aceptadas, cada una con sus defensores y detractores:
 - a. Indicar un tipo NUMBER NOT NULL con sólo dos valores posibles, el '0' y el '1'.
 - b. Indicar un tipo CHAR NOT NULL o similar con sólo dos valores posibles, 'Y' y 'N'.

Para el desarrollo de esta aplicación se escoge la opción a) ya que:

- Es Independiente del idioma del programador.
- Es congruente con las prácticas generalmente usadas para programación en C, C++, Perl, Javascript.
- Encaja mejor para el lenguaje de SQL en la formulación de consultas.
- Aunque ocupa más espacio de almacenamiento no es problema para la dimensión de la base de datos que se está diseñando.

No obstante lo anterior, conviene remarcar que la recomendación de Oracle es el uso de 'Y' y 'N' y las personas que trabajan con Oracle están acostumbradas a verlo de esa manera si no se especifica claramente otra forma.

La definición de las tablas se puede encontrar en el fichero *11_CreacionTablas* que se ha entregado con el código de instalación de la aplicación.

4.3.4 Definición de reglas de la BD

A continuación se detallan las reglas de negocio aplicables para las operaciones de borrado y modificación.

1. En general no se deben permitir las eliminaciones en cascada. Es decir, cuando un dato esté referenciado por otra tabla a través de una clave foránea no se debe permitir su eliminación.

Esta regla sin embargo no se aplica a la eliminación de una tienda o de un producto cuando estén referenciados por una venta. Tal y como se vio durante la fase de análisis de requerimientos, el cliente no quiere guardar los datos relativos a las tiendas cerradas o a los productos que son dados de baja del catálogo. Por tanto se debe permitir la eliminación de una tienda o de un producto pero borrando todos los datos relativos a sus ventas. Esta acción debe motivar la actualización de las estadísticas de dirección.

Para dar solución a este aspecto se permitirá la eliminación en cascada de TIENDAS y PRODUCTOS y se creará un trigger en la tabla VENTAS de modo que se actualicen las estadísticas de dirección cuando se eliminen ventas.

2. Por otra parte, no se debe permitir la modificación de una venta pues se considera una transacción realizada. En caso de que haya que modificar alguna venta por algún tipo de error al introducirla se deberá incluir una venta con un valor de unidades negativo que anule la anterior y volver a introducirla con los valores correctos.

Para dar solución a este punto se creará un trigger en la tabla de VENTAS de modo que no se permita la actualización de datos ya introducidos.

3. Una tienda no puede cambiar su estado de virtual ni de régimen explotación. Sólo se podrá cambiar la dirección postal y el correo del gerente.

La solución será definir un trigger en la tabla TIENDAS que no permita la modificación de los campos FRANQUICIA y VIRTUAL ya introducidos.

4. El número medio de empleados del mes de una tienda puede cambiar a lo largo del mes por lo que habrá que actualizar las estadísticas cuando se modifique dicho valor.

La solución será definir un trigger en la tabla N EMPLEADOS que actualice las estadísticas cuando varíe el número de empleados de una tienda.

5. Dadas las características inherentes a una tienda física, ésta no puede cambiar de ciudad. Sin embargo, sí puede cambiar de ciudad una tienda de venta online.

La solución será definir un trigger en la tabla TIENDAS para controlar dicha modificación.

6. No se puede modificar el código EAN13 de un producto.

La solución será definir un Trigger de modo que se impida la modificación del código EAN13 de un producto.

5. Estructura de la aplicación

5.1 Estructura general

Los ficheros con el código que se ha entregado se estructuran de la siguiente manera:

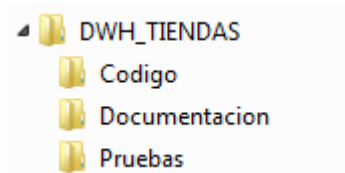


Figura 16: Estructura general del código

En la carpeta *Codigo* se encuentran todos los scripts necesarios para crear la base de datos, creación de Tablespace, de usuarios, tablas, etc.

En la carpeta *Documentacion* está la documentación que acompaña el código en formato HTML para una rápida consulta.

La carpeta *Pruebas* no forma parte de la aplicación pero se incluye para guardar los scripts necesarios para realizar las pruebas. Estos scripts contienen tanto la ejecución de las pruebas como la carga de datos necesarios para que éstas sean ejecutadas.

Se incluye también un fichero *readme.txt* con las instrucciones de instalación de la aplicación.

5.2 Lógica interna

La aplicación consta de 5 paquetes diferentes según la funcionalidad. La información detallada del funcionamiento de los mismos y de los procedimientos y funciones que los conforman será entregada con la documentación de la aplicación dentro de la carpeta *Documentación*.

Todos los procedimientos tienen que registrar su llamada, parámetros de entrada y de salida en la tabla de LOGS. Para ello, siguen el siguiente comportamiento general:

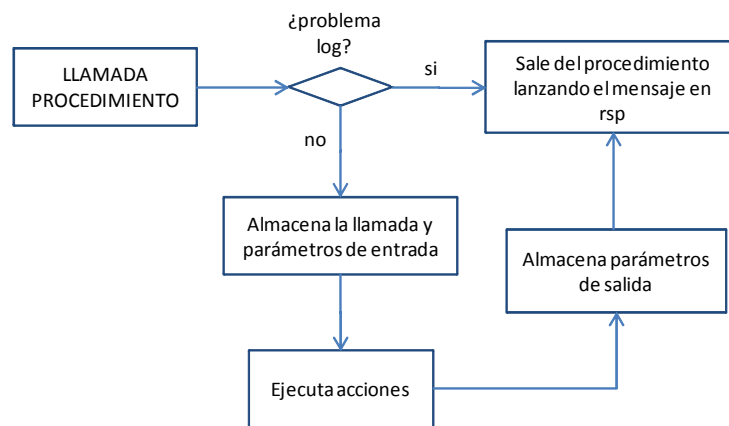


Figura 17: Funcionamiento general interno de un procedimiento

Los procedimientos del Package de control no siguen esta estructura pues sus propias llamadas no son almacenadas en la tabla de LOGS.

Todos los procedimientos tienen un parámetro de salida llamado rsp de tipo STRING que indica si la ejecución ha terminado correctamente ("OK") o si ha terminado con algún tipo de error ("ERROR + Tipo de error").

- **PKG_ADMINISTRACION**

Este paquete contiene los procedimientos para poder realizar las altas, bajas y modificaciones de las tiendas y de los productos. Al no ser un requerimiento del enunciado no se han desarrollado procedimientos para insertar, actualizar o eliminar los datos auxiliares pero en caso de tener que ser desarrollados se incluirían en este paquete.

Los procedimientos de este paquete son los responsables de confirmar los cambios que hayan realizado en la Base de Datos (COMMIT).

- **PKG_CONSULTAS**

Este paquete ofrece los procedimientos y funciones necesarios para poder lanzar las consultas sobre la Base de Datos. Están los procesos tanto para las consultas que corresponden al punto [R4] del enunciado como las consultas relativas a las estadísticas de dirección del punto [R5].

Para las consultas de dirección, se deberá llamar a CONSULTA_ESTADISTICA_GLOBAL, sin indicar ningún ejercicio, si se quiere la estadística con el total de años de existencia de la cadena. Si se necesita un periodo concreto se debe llamar a CONSULTA_ESTADISTICA_ANUAL indicando el periodo para el que se hace la consulta.

Al ser ejecuciones de consultas los procedimientos aquí incluidos no hacen cambios en la Base de Datos (aparte de su propio almacenamiento en la tabla LOGS de lo que se encarga el paquete de control).

- **PKG_CONTROL**

El paquete de control ofrece las funcionalidades para almacenar las llamadas a los procedimientos en la tabla LOGS.

Para ello cuenta con dos procedimientos. El primero para insertar una llamada y los valores de entrada y el segundo para actualizar un log ya creado añadiendo los parámetros de salida. Ambos procedimientos se declaran como "AUTONOMUS_TRANSACCION" para asegurar que los cambios que efectúan sobre la tabla LOGS quedan registrados permanentemente y no se ven afectados por un posible ROLLBACK en la transacción que ha lanzado su ejecución.

- **PKG_EST_DIR**

Sin duda es el más complejo de los cinco paquetes. Es el encargado de mantener actualizada la tabla ESTADISTICAS_DIRECCION donde se almacenan los valores estadísticos los cuales tienen que estar al día en todo momento para poder ser accesibles en tiempo constante mediante una simple SELECT sobre la tabla.

La forma de realizar las actualizaciones de los valores correspondientes a las estadísticas de dirección es a través de dos procedimientos, uno para actualizar la estadística de un año concreto (ACTLZR_EST_ANUAL) y el otro para actualizar la estadística totalizada (ACTLZR_EST_TOTAL). Al insertar una venta se llama a los dos procedimientos. Ambos hacen uso de otro procedimiento que discierne cuál de los dos lo ha llamado para actualizar una u otra fila. **Cada fila de datos estadísticos se identifica con el año al que hace referencia y la fila que almacena los datos agregados se identifica con 0.**

Este último procedimiento en primer lugar busca datos para ese año y si no hubiera inserta la fila haciendo los cálculos con los propios parámetros de la venta que ha provocado su ejecución ya que no habría más ventas para el periodo en cuestión. En caso de que haya datos para ese año, los actualiza convenientemente. Gráficamente, el funcionamiento se resume del siguiente modo:

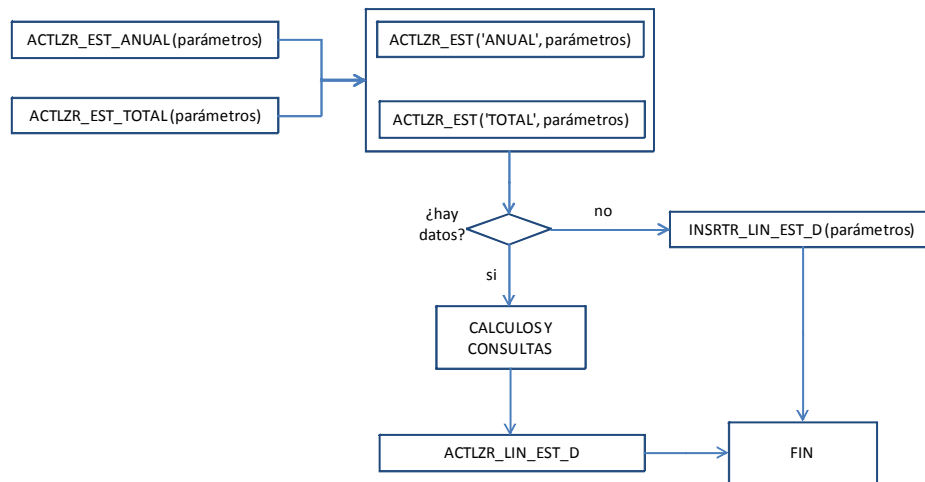


Figura 18: Estructura PKG_EST_DIR

En lo que se refiere a las consultas que se lanzan en tiempo de ejecución, si hay dos valores iguales para una línea, por ejemplo, dos productos en la primera posición de ventas, aparecerá una línea para cada uno de ellos no discriminando ninguno.

En este paquete no se hacen definitivos los cambios en la Base de Datos ya que la responsabilidad de confirmar las transacciones recae sobre los paquetes de Administración, Venta y Control.

- **PKG_VENTA**

Este último paquete es el destinado para uso de los vendedores que han de introducir las ventas en el sistema. El vendedor deberá introducir el producto, la cantidad y la tienda. Los precios vendrán dados por el ERP por lo que para cuando se haga la integración de sistemas habrá que agregar aquí las instrucciones correspondientes para establecer la conexión y toma de datos.

Dado que la fecha para la inserción la genera el sistema automáticamente en el momento de la venta, se ha creado un procedimiento paralelo de pruebas de modo que se puedan hacer inserciones con diferentes fechas para la ejecución de las pruebas de estadísticas.

Los procedimientos de este paquete son los responsables de confirmar los cambios que se han realizado en la Base de Datos (COMMIT).

6. Pruebas

Los ficheros necesarios para ejecutar las pruebas se encuentran en el directorio *DWH_tiendas/Pruebas*.

Las pruebas se realizan en cuatro fases y en un estado en el que la Base de Datos no tiene datos cargados. De otro modo los resultados de las pruebas estadísticas no se pueden comparar con los valores esperados.

En primer lugar se hacen las pruebas para el usuario administrador. Para realizar este grupo de pruebas hay que ejecutar el fichero *200_EjecucionPruebasADM.sql*. Este script establece la conexión para el usuario *DWH_TIENDAS_ADM* y ejecuta los siguientes ficheros:

- *21_DatosAuxiliares.sql* → son los datos auxiliares necesarios para poder realizar las pruebas como Países, Ciudades, etc.
- *22_CargaDatosTest.sql* → prueba el alta de tiendas y productos.
- *23_DatosAuxiliares2.sql* → datos auxiliares que dependen de la existencia previa de tiendas y que no han podido ser cargados anteriormente.

A continuación se puede ejecutar el fichero *201_EjecucionPruebasVEN* que establece la conexión para el usuario *DWH_TIENDAS_VEN* y ejecuta los siguientes ficheros:

- *24_CargaDatosTest2.sql* → prueba el alta de ventas.

La tercera de las fases de pruebas consiste en comprobar los valores estadísticos (los resultados de las mismas se pueden ver a partir de la página 47). Para ello hay que ejecutar, conectado como *DWH_TIENDAS_ADM*, cualquiera de los ficheros denominados como *25_ConsultaEstadisticasn* siendo *n* el número de orden de la estadística.

La última de las fases consiste en la ejecución de otras pruebas como eliminación y modificación de determinados datos así como intentos de alteración de datos que no deben ser modificados. Para lanzar estas pruebas hay que ejecutar desde el usuario *DWH_TIENDAS_ADM* el fichero *26_PruebasUnitarias.sql*.

Cada fichero generará un fichero log con el mismo nombre y en el mismo directorio con los resultados de la ejecución del mismo.

A continuación se muestra el resultado de las pruebas realizadas hasta el momento, organizado por paquetes.

MEMORIA - TFC – Bases de Datos

José Luis Sallent

N.	ENTRADA	ACCION ESPERADA	RDO	VER
PKG_ADMINISTRACION				
1.1	Alta de un producto: PKG_ADMINISTRACION.INSERTAR_PRODUCTO('1234567890123','Falda de punto roja','15/11/2015','Falda','Mujer',rsp);	El producto existe en la BD. La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
1.2	Alta de un producto con un código existente: PKG_ADMINISTRACION.INSERTAR_PRODUCTO('1234567890123','Falda de punto roja','15/11/2015','Falda','Mujer',rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR. Ya existe el producto." como parámetro de salida.	✓	
1.3	Alta de un producto con un tipo equivocado: PKG_ADMINISTRACION.INSERTAR_PRODUCTO('1234567890333','Falda de punto roja','15/11/2015','Falda','Blabla',rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR No existe categoría." como parámetro de salida.	✓	
1.4	Alta de un producto con un género equivocado: PKG_ADMINISTRACION.INSERTAR_PRODUCTO('1234567890333','Falda de punto roja','15/11/2015','Blabla','Mujer',rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR No existe categoría." como parámetro de salida.	✓	
2.1	Modificar la descripción de un producto: PKG_ADMINISTRACION.MODIFICAR_DESCRIPCION_PRODUCTO('1234567890123','Falda de punto verde',rsp);	La tabla de PRODUCTOS recoge el cambio. La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
2.2	Modificar la descripción de un producto que no existe: PKG_ADMINISTRACION.MODIFICAR_DESCRIPCION_PRODUCTO('1234567890333','Falda de punto verde',rsp);	La tabla de LOG almacena la llamada con los parámetros de entrada y "ERROR No existe producto."	✓	
3.1	Eliminar un producto: PKG_ADMINISTRACION.ELIMINAR_PRODUCTO(1234567890123,rsp);	El producto NO EXISTE en la BD.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
		Se actualizan las estadísticas de dirección.	✓	
		Se han borrado sus ventas en la tabla de VENTAS.	✓	
3.2	Eliminar un producto que no existe: PKG_ADMINISTRACION.ELIMINAR_PRODUCTO(1234567890123,rsp);	La tabla de LOG almacena la llamada con los parámetros de entrada y "ERROR. El producto no existe."	✓	
4.1	Alta de una tienda: PKG_ADMINISTRACION.INSERTAR_TIENDA('Barcelona','Cataluña','0,0','rsp);	La tienda existe en la BD.	✓	
		El código de tienda es correlativo con el anterior.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
4.2	Alta de una tienda con un código existente.	El procedimiento de alta no permite dar código manualmente.	✓	
4.3	Alta de una tienda con una localización equivocada: PKG_ADMINISTRACION.INSERTAR_TIENDA('Albacete','Cataluña','0,0','rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR No existe localización." como parámetro de salida.	✓	
5.1	Modificar la dirección de una tienda: PKG_ADMINISTRACION.MODIFICAR_DIRECCION_TIENDA (1,'Ramblas, 27',rsp);	La tabla de TIENDAS recoge el cambio.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
5.2	Modificar la dirección de una tienda que no existe: PKG_ADMINISTRACION.MODIFICAR_DIRECCION_TIENDA (44,'Ramblas, 27',rsp);	La tabla de LOG almacena la llamada con los parámetros de entrada y "ERROR. No existe tienda."	✓	
6.1	Modificar el correo del gerente de una tienda: PKG_ADMINISTRACION.MODIFICAR_CORREO_TIENDA (1,'El correo',rsp);	La tabla de TIENDAS recoge el cambio.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
6.2	Modificar el correo del gerente de una tienda que no existe: PKG_ADMINISTRACION.MODIFICAR_CORREO_TIENDA (44,'El correo',rsp);	La tabla de LOG almacena la llamada con los parámetros de entrada y "ERROR La tienda no existe."	✓	
7.1	Eliminar una tienda: PKG_ADMINISTRACION.ELIMINAR_TIENDA(1,rsp);	La tienda NO EXISTE en la BD.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
		Se actualizan las estadísticas de dirección.	✓	
		Se han borrado sus ventas en la tabla de VENTAS.	✓	
7.2	Eliminar una tienda que no existe: PKG_ADMINISTRACION.ELIMINAR_TIENDA(1,rsp);	La tabla de LOG almacena la llamada con los parámetros de entrada y "ERROR. No existe tienda."	✓	
8.1	Cambiar el valor de una tienda de "Franquicia": UPDATE TIENDAS SET FRANQUICIA = 0 WHERE tienda_id=1;	La aplicación lanza un error con el mensaje "Campos franquicia y virtual no se pueden actualizar."	✓	
8.2	Cambiar el valor de una tienda de "Virtual": UPDATE TIENDAS SET VIRTUAL = 0 WHERE tienda_id=1;	La aplicación lanza un error con el mensaje "Campos franquicia y virtual no se pueden actualizar."	✓	
9.1	Cambiar los empleados del mes de una tienda: UPDATE N_EMPLEADOS SET NUMERO_EMPLEADOS = 7 WHERE TIENDA_ID = 4 AND AÑO = 2015 AND MES = 12;	Se actualizan las estadísticas de dirección.	✓	

Figura 19: Resultado pruebas PKG_ADMINISTRACION

MEMORIA - TFC – Bases de Datos

José Luis Sallent

N.	ENTRADA	ACCION ESPERADA	RDO	VER
PKG_VENTAS				
10.1	Insertar una venta de un producto: PKG_VENTA.INSERTAR_VENTA (11,1234567890134,2,'25,2','10,5',rsp);	La venta existe e la BD.	✓	VER 12.1
		El precio bruto total es igual al precio unitario por las unidades vendidas.	✓	
		El precio beneficio total es igual al beneficio unitario por las unidades vendidas.	✓	
		El número de venta es correlativo con el anterior.	✓	
		La tabla LOG almacena la llamada con los parámetros de entrada y "OK" como parámetro de salida.	✓	
10.2	Insertar una venta de un producto que ya tiene ventas en la misma fecha y hora: PKG_VENTA.INSERTAR_VENTA (11,1234567890134,2,'25,2','10,5',rsp);	La venta existe e la BD.	✓	VER 12.2
		Se actualizan los campos de CTD_TOTAL, PB_TOTAL Y BN_TOTAL.	✓	
10.3	Insertar una venta de un producto que NO tiene ventas en la misma fecha y hora: PKG_VENTA.INSERTAR_VENTA (11,1234567890134,2,'25,2','10,5',rsp);	Se inician los campos de CTD_TOTAL, PB_TOTAL Y BN_TOTAL.	✓	
10.4	Insertar una venta de un producto que NO tiene ventas en la misma fecha y hora pero si hay otros productos vendidos en esa fecha y hora:	Se inician los campos de CTD_TOTAL, PB_TOTAL Y BN_TOTAL.	✓	
10.5	Insertar una venta con una tienda que no existe: PKG_VENTA.INSERTAR_VENTA (20,1234567890134,2,'25,2','10,5',rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR No existe la tienda." como parámetro de salida.	✓	
		No se altera la tabla de Estadísticas de dirección.	✓	
10.6	Insertar una venta con un producto que no existe: PKG_VENTA.INSERTAR_VENTA (17,1234567890244,2,'25,2','10,5',rsp);	La tabla LOG almacena la llamada con los parámetros de entrada y "ERROR No existe el producto." como parámetro de salida.	✓	
		No se altera la tabla de Estadísticas de dirección.	✓	
11.7	Insertar una venta con unidades negativas: PKG_VENTA.INSERTAR_VENTA (17,1234567890123, -2, '25,2', '10,5', rsp);	La venta existe e la BD.	✓	
		Se actualiza la tabla de estadísticas.	✓	

Figura 20: Resultado pruebas PKG_VENTAS

N.	ENTRADA	ACCION ESPERADA	RDO	VER
PKG_EST_DIR				
12.1	Insertar primera venta en la base de datos: PKG_VENTA.INSERTAR_VENTA (11,1234567890134,2,'25,2','10,5',rsp);	Se crea la línea de datos estadísticos para el ejercicio correspondiente a la venta con los datos de la venta.	✓	VER 10.1
		Se crea la línea de datos estadísticos totales correspondiente a la venta con los datos de la venta.	✓	
		La tabla LOG almacena en la tabla LOG la llamada a los procedimientos del paquete utilizados así como los parámetros de entrada y el de salida rsp.	✓	
12.2	Insertar una venta de un producto que ya tiene ventas: PKG_VENTA.INSERTAR_VENTA (11,1234567890134,2,'25,2','10,5',rsp);	Se actualiza la línea de datos estadísticos para el ejercicio correspondiente a la venta con los datos de la venta.	✓	VER 10.2
		Se actualiza la línea de datos estadísticos totales correspondiente a la venta con los datos de la venta.	✓	
		La tabla LOG almacena en la tabla LOG la llamada a los procedimientos del paquete utilizados así como los parámetros de entrada y el de salida rsp.	✓	
12.3	Consultar los valores estadísticos anuales: PKG_CONSULTAS.CONSULTA_ESTADISTICA_ANUAL (2015, listado, rsp);	Los datos son calculados conforme a lo esperado.	✓	VER 13.1
		La tabla LOG almacena en la tabla LOG la llamada a los procedimientos del paquete utilizados así como los parámetros de entrada y el de salida rsp.	✓	
12.4	Consultar los valores estadísticos globales: PKG_CONSULTAS.CONSULTA_ESTADISTICA_GLOBAL (listado, rsp);	Los datos son calculados conforme a lo esperado.	✓	VER 13.2
		La tabla LOG almacena en la tabla LOG la llamada a los procedimientos del paquete utilizados así como los parámetros de entrada y el de salida rsp.	✓	

Figura 21: Resultado pruebas PKG_EST_DIR

MEMORIA - TFC – Bases de Datos

José Luis Sallent

N.	ENTRADA	ACCION ESPERADA	RDO	VER
PKG_CONSULTAS				
13.1	Consultar los valores estadísticos anuales: PKG_CONSULTAS.CONSULTA_ESTADISTICA_ANUAL (2015, listado, rsp);	Si no hay datos de ventas el informe indica 0 para todos los valores.	✓	VER 12.3
		Si hay datos de ventas, el informe muestra los valores esperados.	✓	
		Almacena en la tabla de LOG la llamada al procedimiento, sus parámetros de entrada y de salida de los procedimientos usados (de PKG_CONSULTAS)	✓	
13.2	Consultar los valores estadísticos globales: PKG_CONSULTAS.CONSULTA_ESTADISTICA_GLOBAL (listado, rsp);	Si no hay datos de ventas el informe indica 0 para todos los valores	✓	VER 12.4
		Si hay datos de ventas, el informe muestra los valores esperados	✓	
		Almacena en la tabla de LOG la llamada al procedimiento, sus parámetros de entrada y de salida de los procedimientos usados (de PKG_CONSULTAS)	✓	
13.3	Consultar listado de ventas de tiendas por año y mes: LISTADO_VENTAS_TIENDAS (2015, 11, listado, rsp);	Si no hay valores devuelve un mensaje "ERROR. No ha seleccionado ningún valor."	✓	
		Si hay valores devuelve los valores esperados	✓	
		Almacena en la tabla de LOG la llamada al procedimiento, sus parámetros de entrada y de salida de los procedimientos usados	✓	
13.3	Consultar listado de ventas de productos por año y mes: LISTADO_VENTAS_PRODUCTOS (2015, 11, listado, rsp);	Si no hay valores devuelve un mensaje "ERROR. No ha seleccionado ningún valor."	✓	
		Si hay valores devuelve los valores esperados	✓	
		Almacena en la tabla de LOG la llamada al procedimiento, sus parámetros de entrada y de salida de los procedimientos usados	✓	
13.4	Consultar listado de ventas diarias por año y mes: LISTADO_VENTAS_DIARIAS (2015, 11, listado, rsp);	Si no hay valores devuelve un mensaje "ERROR. No ha seleccionado ningún valor."	✓	
		Si hay valores devuelve los valores esperados	✓	
		Almacena en la tabla de LOG la llamada al procedimiento, sus parámetros de entrada y de salida de los procedimientos usados	✓	

Figura 22: Resultado pruebas PKG_CONSULTAS

N.	ENTRADA	ACCION ESPERADA	RDO	VER
PKG_CONTROL				
Este paquete queda ya testado al hacer las pruebas anteriores.				

Figura 23: Resultado pruebas PKG_CONTROL

MEMORIA - TFC – Bases de Datos

José Luis Sallent

A continuación se muestran los valores esperados de los listados estadísticos conforme a los datos introducidos para la ejecución de las pruebas y se compara con los resultados obtenidos al lanzar las consultas. Los valores esperados se han calculado mediante el uso de tablas dinámicas en Excel.

Listado de ventas por tienda para Noviembre 2015:
25_ConsultaEstadisticas1.sql
LISTADO_VENTAS_TIENDAS (2015, 11, listado, rsp);

Valores esperados:

Tienda	UD	P. DIF.	BENEFICIO N.	%	RATIO EMPLEADO
8	2	1	400	30,75%	40,00
6	2	2	210	16,14%	30,00
10	5	3	132	10,15%	66,00
11	5	5	109	8,38%	109,00
12	5	3	80	6,15%	26,67
4	6	3	72	5,53%	14,40
15	5	3	54	4,15%	13,50
3	3	3	53	4,07%	17,67
14	3	2	32	2,46%	10,67
5	2	2	32	2,46%	5,33
2	1	1	30	2,31%	7,50
16	3	2	28	2,15%	5,60
9	3	3	26	2,00%	8,67
1	2	2	23	1,77%	4,60
17	1	1	10	0,77%	1,67
7	1	1	10	0,77%	1,11

Valores obtenidos:

 ***** LISTADO VENTAS DE TIENDAS *****

TIENDA	UD	P. DIF.	BENEFICIO N.	%	RATIO EMPLEADO
8	2	1	400,00€	30.75	40.00
6	2	2	210,00€	16.14	30.00
10	5	3	132,00€	10.15	66.00
11	5	5	109,00€	8.38	109.00
12	5	3	80,00€	6.15	26.67
4	6	3	72,00€	5.53	14.40
15	5	3	54,00€	4.15	13.50
3	3	3	53,00€	4.07	17.67
14	3	2	32,00€	2.46	10.67
5	2	2	32,00€	2.46	5.33
2	1	1	30,00€	2.31	7.50
16	3	2	28,00€	2.15	5.60
9	3	3	26,00€	2.0	8.67
1	2	2	23,00€	1.77	4.60
17	1	1	10,00€	.77	1.67
7	1	1	10,00€	.77	1.11

Listado de ventas por productos para Noviembre 2015:
25_ConsultaEstadisticas2.sql
LISTADO_VENTAS_PRODUCTOS (2015, 11, listado, rsp);

Valores esperados:

Producto Descripción Tienda	UD VENDIDAS	Beneficio N
1234567890126	3	600
Chaqué top class		
8	2	400
6	1	200
1234567890132	5	150
Zapatos lisos negro de lluvia		
11	1	30
2	1	30
12	1	30
4	1	30
10	1	30
1234567890125	5	150
Camisa de gala		
10	3	90
11	1	30
3	1	30
1234567890130	11	88
Calcetines uso deportivo		
4	4	32
15	3	24
16	2	16
12	2	16
1234567890129	5	85
Blusa beige		
12	2	34
11	1	17
1	1	17
3	1	17
1234567890134	3	60
Chaqueta punto azul		
5	1	20
15	1	20
11	1	20

Producto Descripción Tienda	UD VENDIDA	Beneficio N
1234567890131	5	60
Zapatos tacón		
14	1	12
5	1	12
16	1	12
10	1	12
11	1	12
1234567890133	4	40
Bufanda seda natural		
14	2	20
17	1	10
6	1	10
1234567890123	3	30
Falda de punto roja		
4	1	10
15	1	10
9	1	10
1234567890128	2	20
Falda plisada amarilla		
9	1	10
7	1	10
1234567890127	3	18
Corbata lisa verda		
1	1	6
9	1	6
3	1	6

MEMORIA - TFC – Bases de Datos

José Luis Sallent

Valores obtenidos:

 ***** LISTADO VENTAS DE PRODUCTOS *****

PRODUCTO	DESCRIPCION	UD.	BENEFICIO N.	TIENDA	N.UD
1234567890126	Chaqué top class	3	600,00€	8	2
1234567890125	Camisa de gala	5	150,00€	10	3
1234567890132	Zapatos lisos negro de lluvia	5	150,00€	2	1
1234567890132	Zapatos lisos negro de lluvia	5	150,00€	12	1
1234567890132	Zapatos lisos negro de lluvia	5	150,00€	4	1
1234567890132	Zapatos lisos negro de lluvia	5	150,00€	10	1
1234567890132	Zapatos lisos negro de lluvia	5	150,00€	11	1
1234567890130	Calcetines uso deportivo	11	88,00€	4	4
1234567890129	Blusa beige	5	85,00€	12	2
1234567890131	Zapatos tacón	5	60,00€	14	1
1234567890131	Zapatos tacón	5	60,00€	10	1
1234567890131	Zapatos tacón	5	60,00€	16	1
1234567890131	Zapatos tacón	5	60,00€	5	1
1234567890131	Zapatos tacón	5	60,00€	11	1
1234567890134	Chaqueta punto azul	3	60,00€	15	1
1234567890134	Chaqueta punto azul	3	60,00€	5	1
1234567890134	Chaqueta punto azul	3	60,00€	11	1
1234567890133	Bufanda seda natural	4	40,00€	14	2
1234567890123	Falda de punto verde	3	30,00€	4	1
1234567890123	Falda de punto verde	3	30,00€	9	1
1234567890123	Falda de punto verde	3	30,00€	15	1
1234567890128	Falda plisada amarilla	2	20,00€	9	1
1234567890128	Falda plisada amarilla	2	20,00€	7	1
1234567890127	Corbata lisa roja	3	18,00€	3	1
1234567890127	Corbata lisa roja	3	18,00€	1	1
1234567890127	Corbata lisa roja	3	18,00€	9	1

Listado de ventas diarias para Noviembre 2015:
25_ConsultaEstadisticas3.sql
LISTADO_VENTAS_DIARIAS (2015, 11, listado, rsp);

Valores esperados:

Fecha	Suma de BENEFICIO	Producto	UD	Tienda	Beneficio
15/11/2015	581	1234567890130	7	8	200
16/11/2015	720	1234567890130	4	6	210
16/11/2015	720	1234567890133	4	6	210

Valores obtenidos:

 ***** LISTADO VENTAS DIARIAS *****

FECHA	B. TOTAL	PRODUCTO	UD.	TIENDA	UD
15/11/15	581,00€	1234567890130	7	8	200
16/11/15	720,00€	1234567890133	4	6	210
16/11/15	720,00€	1234567890130	4	6	210

Estadística dirección total
25_ConsultaEstadisticas5.sql
CONSULTA_ÉSTADISTICA_TOTAL (listado, rsp);

Valores esperados:

FECHA, (Todas)		FECHA, (Todas)		FECHA, (Todas)	
Tienda	Suma de BENEFICIO	Producto	Suma de UD_VENDIDAS	Hora	Suma de UD_VENDIDAS
6	464	1234567890130	21	16	20
8	456	1234567890123	12	15	18
12	346	1234567890131	10	10	18
15	294	1234567890125	10	12	16
13	280	1234567890132	10	14	16
10	162	1234567890127	9	13	16
11	147	1234567890128	9	11	12
1	113	1234567890133	8		
4	108	1234567890129	8		
3	101	1234567890134	7		
14	97	1234567890126	7		
5	87	1234567890124	5		
9	81				
2	76				
7	67				
16	62				
17	42				
Total general	2983				
		FECHA, (Todas)			
				Ciudad	Suma de BENEFICIO
				París	1377
				Reus	1121
				Barcelona	485
				Total general	2983
		Virtual	Suma de BENEFICIO		
		no	90,75%		
		si	9,25%		
Fecha	Suma de UD_VENDIDAS				
17/12/2015	29				
15/11/2015	25				
16/11/2015	24				
17/01/2016	21				
10/02/2016	17				

Valores obtenidos:

***** ESTADISTICA ECONOMICA TOTAL DIRECCION *****

```

Año: 0
Beneficio anual: 2983
Tienda de máximas ventas: 6
Beneficio: 464
Producto más vendido: 1234567890130
Unidades: 21
Hora de máximas ventas: 16
Unidades: 20
Hora de menos ventas: 11
Unidades: 12
Día de mayores ventas: 17
Unidades: 50
Día de menores ventas: 10
Unidades: 17
Ciudad de mayores ventas: Paris
Beneficio: 1377
Ratio beneficios tiendas virtuales: 9.25%
    
```

7. Valoración económica

En las empresas dedicadas al desarrollo de software lo habitual es tener unos precios por hora internos. En dicho precio se han de incluir los siguientes componentes:

- un coste medio del personal involucrado en el proyecto según las diferentes categorías. En el caso que nos ocupa voy a considerar las actividades de análisis, diseño, programación y pruebas (PAC2 y PAC3) como evaluables por horas con los siguientes perfiles y coste hora:

Euro

Perfil	Salario medio	Salario supuesto	Coste con carga social	Coste hora
Administrador de proyecto	40.000-45.000	42.000,00	50.400,00	28,00
Analista funcional	33.000-35.000	34.000,00	40.800,00	22,67
Analista programador	32.000-35.000	33.000,00	39.600,00	22,00
Programador	18.000-35.000	21.000,00	25.200,00	14,00

Figura 24: Coste hora por perfil

Para la determinación del coste hora se ha considerado el salario medio publicado en el estudio de remuneración 2015 para el sector de la tecnología de Page Personnel, documento público que se puede encontrar en:

http://www.pagepersonnel.es/productsApp_pp_es/Estudios%20Remuneracion/er_tecnologia.pdf

Para el cálculo se ha estimado que tenemos contratado dicho perfil con un salario supuesto dentro del rango de salario medio, un 20% de incremento por cargas sociales atribuibles a la empresa y 1.800 horas laborables anuales.

Las horas, según la actividad planificada, quedarían distribuidas de la siguiente manera:

Id	Previa	Tarea	Horas	Administrador de proyecto	Analista funcional	Analista programador	Programador
2	1	PAC2	60,5				
2.1		Análisis de requerimientos	12		12		
2.2	2.1	Redacción de documento de especificación	4		4		
2.3	2.1	Diseño conceptual	32		32		
2.4	2.2, 2.3	Redacción de documento a entregar	12	4	8		
2.5	2.4	Entrega de PAC 2	0,5	0,5			
3	2	PAC3	52,5				
3.1		Diseño lógico	12			12	
3.2	3.1	Programación de Base de datos	20				20
3.3	3.2	Redacción de documento a entregar	6			6	
3.4	3.2	Elaboración de los grupos de pruebas	6				6
3.5	3.4	Ejecución de las pruebas y retoques de código	4				4
3.6	3.2, 3.5	Preparar documentos final para entrega	4	2		2	
3.7	3.6	Entrega PAC 3	0,5			0,5	
			Total horas	6,5	56	20,5	30

Figura 25: Consumo de horas planificadas según perfil

Así por tanto, el coste de las horas empleadas quedaría como sigue:

	Administrador de proyecto	Analista funcional	Analista programador	Programador
Total horas	6,5	56	20,5	30
Total Euro	182,00 €	1.269,33 €	451,00 €	420,00 €
Suma Total:				2.322,33 €

Figura 26: Coste total planificado de horas empleadas

- un componente de los costes estimados de administración y dirección. A estos efectos los costes de las actividades de calificación y memoria (PAC1 y entrega final) se entenderán imputados ya en esta repercusión. Haciendo el supuesto de que hay un histórico de datos económicos en la empresa se puede estimar un 6% de los costes incurridos.
- un componente de los costes de amortización y financieros. Habrá que estimar también sobre un histórico supuesto. Lo dejaré en un 2%.
- un margen de beneficio económico. Imaginamos que la empresa quiere trabajar con un 20% de margen sobre ventas.

Con los datos anteriores se determinó en fase de planificación un precio de venta de 3.135 Euro tal y como se desglosa en la siguiente tabla.

Suma Anterior	2.322,33 €	
Coste Administración:	139,34 €	6%
Coste amort. y finan.:	46,45 €	2%
Margen:	627,03 €	20%
Venta:	3.135,15 €	

Figura 27: Determinación del precio de venta inicial

Como se ha visto en el punto 2.8, ha habido una desviación de 55 horas. Lógicamente esto ha de tener una repercusión en la valoración económica. Para analizar el impacto habrá que ver las horas consumidas según el perfil de cada empleado. Las horas reales han quedado distribuidas según se muestra en la figura 28

MEMORIA - TFC – Bases de Datos

José Luis Sallent

Id	Previa	Tarea	Administrador de proyecto	Analista funcional	Analista programador	Programador
2	1	PAC2				
2.1		Análisis de requerimientos		21		
2.2	2.1	Redacción de documento de especificación		4		
2.3	2.1	Diseño conceptual		17		
2.4	2.3	Diseño lógico			3	
2.5	2.2, 2.4	Redacción de documento a entregar	4	6		
2.6	2.5	Entrega de PAC 2	0,5			
3	2	PAC3				
3.1		Diseño físico			13,5	
3.2	3.1	Programación de Base de datos				68,5
3.3	3.2	Redacción de documento a entregar			6	
3.4	3.2	Elaboración de los grupos de pruebas				4
3.5	3.4	Ejecución de las pruebas y retoques de código				12
3.6	3.2, 3.5	Preparar documentos final para entrega	2		4	
3.7	3.6	Entrega PAC 3			0,5	
			6,5	48	27	84,5

Figura 28: Consumo de horas reales según perfil

Con estos consumos y los costes expuestos en la figura 24, el coste para las fases intermedias sería

	Administrador de proyecto	Analista funcional	Analista programador	Programador
Total horas	6,5	48	27	84,5
Total Euro	182,00 €	1.088,00 €	594,00 €	1.183,00 €

Figura 29: Coste total real de horas empleadas

Por tanto, el precio de venta quedaría como se indica a continuación:

Suma Anterior	3.047,00 €
Coste Administración:	182,82 €
Coste amort. y finan.:	60,94 €
Margen:	822,69 €
Venta:	4.113,45 €

Figura 30: Determinación del precio de venta final

El cambio en la planificación supone un sobrecoste de 978,3 Euros respecto de los 3.135,15 Euros que se había estimado en la fase de calificación del proyecto. Esto significa que la desviación en la fase de programación, como se dice normalmente, se “ha comido” el margen previsto y provoca que si queremos mantener el precio, se cierre la venta con margen negativo sobre ventas del 5%.

8. Conclusiones

Tras todo el trabajo realizado a lo largo del semestre cabe preguntarse si los objetivos iniciales se han alcanzado. Si se vuelve al punto 2.2 y se revisan los objetivos planteados puedo afirmar que éstos se han conseguido.

Me gustaría destacar como este trabajo ha puesto de relieve la importancia de una buena planificación, no sólo para la obtención de un producto en plazo sino también en su vertiente económica. He querido mostrar a lo largo de estas líneas como algunos errores de estimación en el consumo de tiempos pueden llevar a una empresa en una situación real a perder los márgenes esperados llegando incluso a márgenes netos negativos, lo que a la larga la llevaría al cierre.

Por otra parte he podido experimentar la importancia de dominar los diferentes sistemas y lenguajes de programación. Estoy convencido de que personas con experiencia previa en Oracle hubieran realizado la programación en tiempos muchísimo más reducidos y probablemente de forma más apropiada. No obstante, he aprendido una herramienta nueva lo que sin duda sólo puede ser positivo. En particular, no conocía hasta ahora el uso de cursores, algo que encuentro muy útil y a lo que sin duda dedicaré un tiempo de estudio para sacarle el máximo provecho.

Por último me gustaría comentar que el hecho de llevar a cabo todas las fases de un proyecto informático me ha permitido ver en conjunto las fases que de forma individual he ido estudiando a lo largo de la carrera. Si duda es un buen trabajo de síntesis de conocimientos.

9. Glosario

Base de Datos: Conjunto de información estructurada de forma normalizada, almacenada en un dispositivo y entre la que se establece un número limitado o ilimitado de relaciones.

Clave primaria: Atributo o combinación de atributos que identifica de forma única a cada instancia de una entidad.

Diseño conceptual: Conjunto de actividades encaminadas a definir la estructura de una Base de Datos partiendo de los requerimientos y sin considerar el Sistema Gestor de Bases de Datos a emplear.

Diseño físico: Etapa en el desarrollo de una Base de Datos para producir la descripción de la implementación de la Base de Datos.

Diseño lógico: Conjunto de actividades encaminadas a adaptar un diseño conceptual de una Base de Datos a la tecnología que se va a emplear para su desarrollo.

ERP (Enterprise Resource Planning): Sistema informático para la ejecución y gestión de procesos de negocio asociados a las operaciones de una empresa.

Franquicia: Concesión de los derechos de explotación de una actividad otorgada por una empresa a una o varias personas en una zona determinada.

Modelo Entidad – Relación (E-R): Es un modelo basado en una serie de conceptos para describir la estructura de una Base de Datos en un lenguaje de alto nivel y que es independiente de la tecnología a emplear para su desarrollo.

Sistema de Gestión de Bases de Datos: Software para la utilización, actualización y manipulación de datos almacenados en una o varias Bases de Datos.

Tienda física: Casa, puesto o lugar donde existe un espacio físico destinado a la venta al público artículos de comercio al por menor.

Tienda virtual: Lugar donde se venden al público artículos de comercio al por menor sin que exista espacio un físico destinado a la exposición y venta.

10. Bibliografía

Alberto Gómez Gómez, Nicolás de Abajo Martínez (1998). *Los sistemas de información en la empresa*. Ed. Servicio de publicaciones Universidad de Oviedo.

Barceló García, Miquel, *El proyecto informático de construcción de software*. Gestión y Organización de Proyectos informáticos. Universitat Oberta de Catalunya.

Barceló García, Miquel, *La gestión de un proyecto informático*. Gestión y Organización de Proyectos informáticos. Universitat Oberta de Catalunya.

Campderrich Falgueras, Benet, *Introducción a la ingeniería del software OO*. Ingeniería del software. Universitat Oberta de Catalunya.

Campderrich Falgueras, Benet, *Recogida y documentación de requisitos*. Ingeniería del software. Universitat Oberta de Catalunya.

Campderrich Falgueras, Benet y Recerca Informàtica, S.L., *UML (II): El modelo dinámico y de implementación*. Ingeniería del software. Universitat Oberta de Catalunya.

Costal Costa, Dolors, *Diseño de Bases de Datos – Modulo 5*. Bases de Datos I. Universitat Oberta de Catalunya.

Doug Rosenberg and Matt Stephens (2007). *Use Case Driven Object Modeling with UML. Theory and Practice*. Ed. Apress.

Laurent Debrauwer y Fien Van der Heyde (2005). *UML2, Iniciación, ejemplos y ejercicios corregidos*. Ed. Ediciones ENI. Edición Española: Andrea García Vega.

Pla docent - TFC - Bases de dades relacionals (set.15 – feb. 16). Universitat Oberta de Catalunya.

Unified Modeling Language™ (UML®) Resource Page
<http://www.uml.org/>

11. Anexo I. Correspondencia con el cliente

17/10/2015

Buzón



Asunto: **Aclaraciones sobre requerimientos funcionales**

De: Jose Luis Sallent Sanchez Añadir contacto

A: Foro TFC-BD Relacionals aula 1

Copia a:

Fecha: 17.10.2015 21:12

Buenas tardes,

Tras leer detalladamente los requerimientos expuestos en el enunciado me surgen las siguientes dudas las cuales serían planteadas en una reunión con el cliente para precisarlas correctamente. Lanzo las cuestiones en el fórum por si son de utilidad para algún compañero de aula. Haré las preguntas siguiendo el orden del enunciado para mayor facilidad.

[R2]

Respecto del código EAN13 del producto, ¿hay que determinar la validez del número mediante el dígito de control del mismo o hay que entender que ya es válido el código introducido?

[R3]

En la página 3 al referirse a

"cantidad total del producto vendido ese día y hora",
"precio bruto total de los productos vendidos aquel día y hora",
"beneficio neto total de los productos vendidos aquel día y hora",

¿se refiere al producto identificado con ese número EAN13 que corresponde a la venta o se refiere a todos los productos vendidos? Por favor, identifica cada caso concreto si las respuestas son diferentes a cada ratio.

[R4]

En cuanto a los procedimientos de ABM para la información de ventas quisiera aclarar el punto de vista. En mi opinión no debería permitirse modificación o una baja de una venta. Si se permitiera podría dar lugar a manipulaciones malintencionadas de las estadísticas y por tanto parece más lógico que la venta no pueda alterarse y, en todo caso, aceptar una venta "negativa" para poder registrar una devolución o una corrección. De cara al futuro sería interesante para otro tipo de estadísticas.

En los procedimientos de consulta a), el "número de productos vendidos ese mes", ¿se refiere al número de unidades de todos los productos vendidos en ese periodo o al número de ventas de ese mes, al margen de la unidades de cada venta?

Otros aspectos:

En lo que se refiere al precio de venta quisiera aclarar si cada tienda tiene libertad para establecer sus precios o si cada producto tiene un precio fijado al que todas las tiendas tienen que ceñirse.

Por último, en uno de los puntos del enunciado se pide que el sistema muestre en un listado el "beneficio neto dividido por el número de empleados de la tienda". Dado que por una parte se pide que la consulta sea dinámica en tiempo de ejecución y por otra parte el número de empleados es una magnitud que varía con el tiempo, se podrían generar estadísticas que llevaran a la confusión y que dieran valores diferentes para meses ya cerrados si varía el número de empleados. Por tanto o bien se genera un histórico del número de empleados por tienda y mes o bien se establece la premisa de que el número de empleados es una magnitud fija en el tiempo. ¿Qué postura tiene el cliente?

Si alguna pregunta no se entiende bien dímelo y la reformulo.

Gracias y un cordial saludo

José Luis Sallent



Asunto: **RE: Aclaraciones sobre requerimientos funcionales**

De: [Juan Martínez Bolaños](#)

A: Foro TFC-BD Relacionales aula 1

Copia a:

Fecha: 19.10.2015 20:11

Hola José Luis,

Podemos asumir que el código EAN13 será correcto.
Sí, se refiere al producto identificado con el número EAN13.
Me parece bien tu razonamiento. Explícalo en la memoria y implementa el aceptar una venta "negativa".
Se refiere al número de unidades de todos los productos.
Los precios vendrían dados por el ERP. Se prefiere que el número de empleados pueda variar con el tiempo.

Cualquier cosa que no quede suficientemente clara, me comentas y lo aclaramos.

Saludos

23/10/2015

Buzón



Asunto: **Dudas de gestión de negocio**
De: Jose Luis Sallent Sanchez Añadir contacto
A: Foro TFC-BD Relacionals aula 1
Copia a:
Fecha: 22.10.2015 20:13

Buenas noches,

De cara a una mejor implementación de la solución me gustaría obtener confirmación sobre algunos aspectos que se derivarían de la forma de gestión de la cadena de tiendas. Además, me gustaría conocer el interés del cliente en determinados datos adicionales. (Estas dudas se plantearían en una revisión de requisitos con el cliente).

En cuanto a gestión y en particular en lo que se refiere a los gerentes, entiendo que una tienda sólo puede tener un gerente pero, ¿se podría dar el caso de dos tiendas dirigidas por la misma persona? ¿Existe la posibilidad de que la gerencia esté vacante de forma temporal?

Por otra parte necesitaría saber si la cadena está interesada en incluir los datos completos de localización de las tiendas, incluyendo el País de cara a una futura internacionalización así como la dirección postal. No supone un coste adicional en este momento de desarrollo. Relativo a las tiendas también, ¿se prevé que se pueda cerrar una tienda? En caso de cierre, ¿se quiere mantener la información relativa a sus ventas?

Por la vertiente de los productos, igual que antes, incluir alguna categorización de los productos en el catálogo no lleva coste adicional y puede ser positivo de cara a futuras estadísticas por grupos de productos. Así mismo, ¿se prevé que se pueda dar de baja un producto? En caso afirmativo, ¿se quiere mantener la información relativa a sus ventas?

Por la vertiente de los productos, igual que antes, incluir alguna categorización de los productos en el catálogo no lleva coste adicional y puede ser positivo de cara a futuras estadísticas por grupos de productos. Así mismo, ¿se prevé que se pueda dar de baja un producto? En caso afirmativo, ¿se quiere mantener la información relativa a sus ventas?

Por último, ¿hace el cliente una gestión de retribuciones variables a los vendedores en función de las ventas? Este sistema le podría ser de utilidad en un futuro.

Gracias de antemano por las respuestas

José Luis Sallent



Asunto: **RE: Dudas de gestión de negocio**
De: Juan Martínez Bolaños
A: Foro TFC-BD Relacionals aula 1
Copia a:
Fecha: 23.10.2015 07:52

Hola,
Contesto a tus dudas:

Un gerente diferente por tienda y puede que esté vacante de forma temporal.
Sí, queremos los datos de localización de las tiendas. En caso de cerrar tienda no hace falta conservar los datos.
Me parece bien la categorización de productos. Como en el caso anterior, en casos de dar de baja un producto no hace falta mantener su información.
No, no hay retribuciones variables.

Saludos