

Sistema de control de registres sanitaris

Memòria

El Treball fi de grau (TFG) és una assignatura que està pensada per a realitzar un treball de síntesi dels coneixements adquirits en altres assignatures de la carrera i que requereixi posar-los en pràctica conjuntament en un treball concret.

Disseny i implementació d'un sistema de control de registres sanitaris

Jaume López i Bonells
Grau d'Enginyeria Informàtica

Jordi Ferrer Duran

12 de juny de 2013

Aquest treball està dedicat de tot cor als meus pares, la Victòria i a tota la meva família, per haver-me engrescat sempre en tots els projectes personals i professionals que he decidit emprendre.

Dedicat especialment al meu germà, Jordi López i Bonells; si no fos per ell aquest treball i la meva carrera com a informàtic, no hauria estat possible.

1 Resum

Aquest Treball fi de grau (TFG), inclòs a l'àrea de Bases de Dades, presenta un problema del món real en el qual s'han de demostrar les competències adquirides a les assignatures de *Ús de bases de dades*, *Disseny de bases de dades* i *Arquitectura de bases de dades*, entre d'altres.

El treball proposat consisteix en implementar un sistema de BD per a donar resposta a la necessitat d'una gran empresa del sector cosmètic que ens ha contactat ja que necessita dissenyar un sistema de control dels registres sanitaris que ha de realitzar en els diferents països a on es comercialitzen els seus productes. Per raons sanitàries, abans de poder començar a vendre cap producte cosmètic en un país determinat es necessita passar un conjunt de tràmits burocràtics que s'anomenen registre.

En aquest cas, l'abast del TFG comprèn la planificació, anàlisi, disseny, implementació i proves del sistema. Tota la gestió i accés a la informació es farà mitjançant procediments de Bases de Dades, sent aquesta l'única manera d'accedir-hi. Per tant, no caldrà implementar cap interfície d'usuari amb un llenguatge de programació d'alt nivell.

Partint dels requisits del client, aquesta BD ha de permetre emmagatzemar tota la informació necessària per a modelar el sistema de control dels registres sanitaris. S'han de guardar dades sobre productes, com poden ser el seu format i presentació, dades dels tràmits burocràtics, tals com a on es fa el registre i quin distribuïdor homologat realitzarà la venda, així com dades sobre països, moneda, etc.

D'altra banda, també ens demanen algunes funcionalitats com l'alta, baixa i modificació dels elements esmentats anteriorment, així com la generació de dades estadístiques que, juntament amb les funcionalitats anteriors facilitarà les actuacions de millora del control de registres. Aquesta informació s'ha de generar automàticament a mesura que es produeixen actualitzacions al sistema.

2 Índex

2.1 Índex de continguts

1	RESUM	2
2	ÍNDEX	3
2.1	ÍNDEX DE CONTINGUTS	3
2.2	ÍNDEX DE TAULES	4
2.3	ÍNDEX DE FIGURES	4
3	INTRODUCCIÓ	5
3.1	JUSTIFICACIÓ DEL TFG I CONTEXT EN EL QUAL ES DESENVOLUPA	5
3.2	OBJECTIUS DEL TFG	5
3.2.1	<i>Objectius generals</i>	5
3.2.2	<i>Competències</i>	6
3.2.3	<i>Objectius específics</i>	6
3.3	ENFOCAMENT I MÈTODE SEGUIT	6
3.4	PLANIFICACIÓ DEL TREBALL	7
3.4.1	<i>Tasques</i>	7
3.4.2	<i>Planificació temporal</i>	8
3.4.3	<i>Diagrama de Gantt</i>	9
3.5	PRODUCTES OBTINGUTS	10
3.5.1	<i>Lliuraments</i>	10
3.5.2	<i>Documentació parcial</i>	10
3.5.2.1	Planificació TFG (PAC 1: Pla de Treball)	10
3.5.2.2	Anàlisi i Disseny (PAC 2)	10
3.5.2.3	Implementació de la Base de Dades (PAC 3)	10
3.5.2.4	Elaboració Memòria i Presentació virtual (LLIURAMENT FINAL)	10
3.6	BREU DESCRIPCIÓ DELS ALTRES CAPÍTOLS DE LA MEMÒRIA	11
4	ANÀLISI DE REQUISITS	12
4.1	DESCRIPCIÓ INICIAL	12
4.2	REQUISITS FUNCIONALS	12
4.3	MÒDUL ESTADÍSTIC	14
4.4	LOGS	14
5	DISSENY	15
5.1	DISSENY CONCEPTUAL	15
5.1.1	<i>Identificació d'entitats i atributs</i>	15
5.1.1.1	Tipus d'entitats	15
5.1.1.2	Atributs de les entitats	16
5.1.1.3	Relació de les entitats	18
5.1.2	<i>Diagrama entitat/relació</i>	19
5.1.3	<i>Restriccions d'integritat</i>	20
5.2	DISSENY LÒGIC	20
5.2.1	<i>Model relacional</i>	20
5.3	DISSENY FÍSIC	21
5.3.1	<i>Instal·lació i configuració del SGBD Oracle</i>	21
5.3.2	<i>Creació de la Base de Dades</i>	22
5.3.2.1	Espai per a taules (tablespaces)	22
5.3.2.2	Usuaris	23
5.3.3	<i>Creació de taules</i>	24
5.3.3.1	Taules del mòdul principal	24
5.3.3.2	Taules del mòdul estadístic	27
5.3.4	<i>Índexs</i>	28

6	IMPLEMENTACIÓ I PROVES	29
6.1	PARELLS SEQÜÈNCIA – DISPARADOR	29
6.2	IMPLEMENTACIÓ DE FUNCIONALITATS	29
6.2.1	<i>Procediments d’alta, baixa i modificació (ABM)</i>	29
6.2.2	<i>Mòdul estadístic</i>	38
6.2.3	<i>Logs</i>	41
6.3	PROVES DEL SISTEMA	42
6.3.1	<i>Càrrega inicial de dades</i>	42
6.3.2	<i>Proves de procediments ABM</i>	43
6.3.3	<i>Proves del mòdul estadístic</i>	44
6.3.4	<i>Comprovació de logs</i>	48
7	RECURSOS EMPLEATS I VALORACIÓ ECONÒMICA	49
7.1.1	<i>Recursos humans</i>	49
7.1.2	<i>Maquinari</i>	49
7.1.3	<i>Programari</i>	49
7.1.4	<i>Valoració econòmica</i>	49
8	CONCLUSIONS	51
9	GLOSSARI	52
10	BIBLIOGRAFIA	55
10.1	MATERIALS DIDÀCTICS D’ALTRES ASSIGNATURES	55
10.2	LLIBRES DE CONSULTA	55
10.3	ENLLAÇOS D’INTERNET	55
11	ANNEXOS	56
11.1	MANUAL DE FUNCIONAMENT	56
11.2	VALORACIÓ ECONÒMICA DELS RECURSOS	58
11.3	INSTAL·LACIÓ I CONFIGURACIÓ DEL SGBD ORACLE	58

2.2 Índex de taules

Taula 1:	Dates clau	7
Taula 2:	Tasques	8
Taula 3:	Regles de negoci per a la Base de Dades	12
Taula 4:	Punts oberts dels requisits funcionals	14
Taula 5:	Relacions entre entitats	18
Taula 6:	Càrrega de dades	42
Taula 7:	Relació Perfil Professional – Cost – Jornada	50
Taula 8:	Valoració econòmica	50
Taula 9:	Cost del maquinari	58
Taula 10:	Cost del programari	58

2.3 Índex de figures

Figura 1:	Diagrama de Gantt	9
Figura 2:	Model E/R de la Base de Dades	19

3 Introducció

3.1 Justificació del TFG i context en el qual es desenvolupa

El Treball fi de grau és una assignatura que està pensada per a realitzar un treball de síntesi dels coneixements adquirits en altres assignatures de la carrera i que requereixi posar-los en pràctica conjuntament en un treball concret. Normalment el TFG és un treball eminentment pràctic i vinculat a l'exercici professional de la informàtica encara que en alguns casos pot ser, o incloure, un treball de recerca.

Centrat a l'àrea de bases de dades relacionals, aquest TFG pretén oferir a l'estudiant l'oportunitat de posar en pràctica el coneixements adquirits emprant algun sistema de gestió de bases de dades relacional dels que hi ha al mercat laboral. A l'enunciat es demana que es realitzi el treball utilitzant el sistema de Gestió de Bases de Dades (SGBD) Oracle, el qual gaudeix de molt bona posició a nivell de mercat, fet que justifica la seva utilització.

El punt de partida és la petició per part d'una gran empresa del sector cosmètic d'un sistema de Bases de Dades (BD) que ens ha contactat ja que necessita dissenyar un **sistema de control de registres sanitaris** que ha de realitzar en els diferents països a on es comercialitzen els seus productes. A partir d'uns requisits inicials es realitzarà l'anàlisi, el disseny i la implementació d'un sistema de BD que donarà suport a una futura aplicació que permetrà generar estadístiques necessàries per determinar actuacions i plans de millora.

3.2 Objectius del TFG

La informació ha esdevingut un recurs d'importància primordial a totes les organitzacions, independentment de quins siguin els seus àmbits de negoci i actuació. Per això, les organitzacions inverteixen grans esforços en el desenvolupament de sistemes d'informació per a la producció i gestió de la informació.

El model relacional fou proposat el 1970 per E.F. Codd. Les bases de dades relacionals i els sistemes gestors de bases de dades que hi donen suport són, amb diferència, el tipus de bases de dades actualment més difós i utilitzat. El principal motiu d'aquest èxit és que ofereixen sistemes simples i eficaços per a representar i manipular dades en un ventall ampli d'entorns d'aplicació, a la vegada que es preserva la integritat de les dades emmagatzemades.

Conseqüentment, aquesta àrea de TFG pretén donar l'oportunitat a l'estudiant de consolidar i ampliar els coneixements adquirits en l'àrea de bases de dades durant els Estudis i posar-los en pràctica emprant algun sistema de gestió de bases de dades relacional.

3.2.1 Objectius generals

Els objectius que es persegueixen en el desenvolupament d'aquest treball són els següents:

- Posar en pràctica els coneixements adquirits en assignatures de Bases de Dades.
- Emprar el llenguatge SQL.
- Ampliar coneixements utilitzant noves eines.
- Detectar quines són les necessitats bàsiques d'un determinat sistema a analitzar.
- Detectar possibles funcionalitats addicionals de valor afegit.
- Proposar un disseny que s'ajusti als requeriments exposats.
- Implementar un sistema que encapsuli les funcions d'accés a les dades.

Per tant, és imprescindible haver cursat les assignatures obligatòries que corresponen amb les matèries de Bases de Dades i Enginyeria del Programari. Aquestes assignatures són, respectivament, Disseny de Bases de Dades i Enginyeria del Programari. També és convenient haver cursat o estar cursant en el mateix semestre l'assignatura de Gestió de Projectes.

3.2.2 Competències

Dins del TFG es treballen les competències transversals següents, ja desenvolupades al llarg del Grau:

- Capacitat de comunicació escrita en l'àmbit acadèmic i professional.
- Ús i aplicació de les TIC en l'àmbit acadèmic i professional.
- Capacitat per adaptar-se a les tecnologies i als futurs entorns actualitzant les competències professionals.
- Capacitat per innovar i generar noves idees.

També es treballen les competències específiques següents del Grau:

- Capacitat per planificar i gestionar projectes en l'entorn de les TIC.
- Capacitat per avaluar solucions tecnològiques i elaborar propostes de projectes tenint en compte els recursos, les alternatives disponibles i les condicions de mercat.

3.2.3 Objectius específics

En aquesta àrea es treballaran aspectes directament relacionats amb l'especificació, el disseny i la implementació de bases de dades donats uns requisits d'informació i funcionals. Es pretén que mitjançant la realització d'un projecte de complexitat mitjana s'apliquin les tècniques i se segueixin les metodologies estudiades durant els Estudis amb el rigor metodològic necessari.

Des d'un punt de vista d'implementació de la base de dades, es faran els scripts necessaris per tal de crear la base de dades i implementar els procediments emmagatzemats i disparadors necessaris per a què el sistema funcioni i els aplicatius els utilitzin. També es poden tractar aspectes de magatzems de dades (data warehouse).

3.3 Enfocament i mètode seguit

El primer pas per afrontar aquest projecte ha estat realitzar una anàlisi inicial dels requeriments recollits en l'enunciat i elaborar un pla realista i ajustat que permeti aconseguir en termini, abast i qualitat els objectius establerts en apartats anteriors.

Per aconseguir una estimació ajustada del volum de treball i una planificació el més realista possible, s'ha estimat convenient utilitzar el **cicle de vida clàssic (o en cascada)**, el qual és útil en projectes amb una durada fitada i on sigui possible definir clarament uns requisits inicials que no vagin canviant massa al llarg del mateix.

Aquest mètode de desenvolupament de projectes distingeix diverses fases i es caracteritza per l'ordre lineal d'aquestes. Les fases que es preveuen en el nostre cas són les següents:

- Anàlisi de requisits: estudi detallat dels requeriments i necessitats que haurà de resoldre el futur programari. En aquesta fase és molt important la comunicació amb el client amb l'objectiu de concretar al màxim les seves necessitats.
- Disseny: s'especifica una solució als requeriments obtinguts a la fase anterior i se'n defineixen les estructures necessàries.
- Implementació: és la fase a un s'implementa el codi font.
- Prova: es prova el producte de manera planificada amb l'objectiu de detectar i corregir possibles errades.
- Manteniment: Durant el temps d'ús del producte es fan els canvis necessaris per tal de corregir errors o per adaptar-lo a noves necessitats.

Aquest cicle de vida es l'indicat quan no es preveuen molts canvis sobre els requisits inicials durant la resta del projecte. Aquest fet permet fer una bona planificació i valoració econòmica des del començament.

3.4 Planificació del treball

La **data inicial** del projecte correspon a la data del lliurament de l'enunciat (28/02/2013) i la **data final** del projecte correspon a la data del lliurament final del producte, la memòria i la presentació virtual (12/06/2013). Les dates clau del treball són les següents:

Taula 1: Dates clau

Esdeveniment	Data
Inici	28/02/2013
Lliurament PAC 1: Pla de Treball	17/03/2013
Lliurament PAC 2: Anàlisi i Disseny	21/04/2013
Lliurament PAC 3: Implementació	19/05/2013
Entrega final: producte, memòria i presentació	12/06/2013
Inici Debat Virtual	25/06/2013
Final Debat Virtual	28/06/2013

3.4.1 Tasques

Planificació:

- Lectura de l'enunciat, identificació de tasques a realitzar i planificació temporal
- Elaboració de la documentació (PAC1: Pla de Treball)

Anàlisi i disseny:

- Revisió de requisits
- Definició detalla dels requisits inicials
- Comunicació amb el client per a resoldre possibles dubtes
- Actualització dels requisits
- Disseny conceptual
- Identificació d'entitats de l'enunciat
- Identificació d'entitats d'estadístiques i log
- Diagrames UML o E/R
- Disseny (lògic i físic)
- Elaboració de la documentació (PAC2: Anàlisi i Disseny de la BD)

Implementació i proves:

- Instal·lació i familiarització Oracle
- Implementació del requisits bàsics
- Creació BD i taules (disparadors i procediments ABM)
- Implementació d'estadístiques i log
- Elaboració Jocs de proves (i modificacions derivades d'aquests)
- Elaboració de la documentació (PAC3: Implementació de la BD)

Documentació:

- Elaboració de la memòria i de la presentació virtual

3.4.2 Planificació temporal

Taula 2: Tasques

<i>Tasca</i>	<i>Duració</i>	<i>Inici</i>	<i>Final</i>
TFC Bases de Dades	107 dies	28/02/13	28/06/13
Lectura de l'enunciat	2 dies	28/02/13	01/03/13
01 - Planificació TFG	13 dies	02/03/13	17/03/13
01.01 - Redacció resum i objectius	2 dies	02/03/13	04/03/13
01.02 - Identificació de tasques a realitzar	1 dia	05/03/13	05/03/13
01.03 - Planificació temporal	1 dia	06/03/13	06/03/13
01.04 - Elaboració Pla de Treball	9 dies	07/03/13	16/03/13
01.05 - Lliurament Pla de Treball (PAC 1)	0 dies	17/03/13	17/03/13
02 - Anàlisi i Disseny	30 dies	18/03/13	21/04/13
02.01 - Revisió de requisits	9 dies	18/03/13	27/03/13
02.01.01 - Definició detalla dels requisits inicials	4 dies	18/03/13	21/03/13
02.01.02 - Comunicació amb el client	3 dies	22/03/13	25/03/13
02.01.03 - Actualització dels requisits	2 dies	26/03/13	27/03/13
02.02 - Disseny	16 dies	28/03/13	15/04/13
02.02.01 - Disseny conceptual	9 dies	28/03/13	06/04/13
02.02.01.01 - Identificació d'entitats de l'enunciat	3 dies	28/03/13	30/03/13
02.02.01.02 - Identificació d'entitats d'estadístiques i log	2 dies	01/04/13	02/04/13
02.02.01.03 - Diagrames UML o E/R	4 dies	03/04/13	06/04/13
02.02.02 - Disseny lògic	5 dies	08/04/13	12/04/13
02.02.03 - Disseny físic	2 dies	13/04/13	15/04/13
02.03 - Elaboració de la documentació	5 dies	16/04/13	20/04/13
02.04 - Lliurament Documentació (PAC 2)	0 dies	21/04/13	21/04/13
03 - Implementació i proves	25 dies	22/04/13	19/05/13
03.01 - Instal·lació SGBD Oracle	1 dia	22/04/13	22/04/13
03.02 - Implementació del requisits bàsics	13 dies	23/04/13	07/05/13
03.02.01 - Creació BD i taules	2 dies	23/04/13	24/04/13
03.02.02 - Creació parells seqüència-disparadors	2 dies	25/04/13	26/04/13
03.02.03 - Creació procediments ABM	4 dies	27/04/13	01/05/13
03.02.04 - Procediments de consulta	5 dies	02/05/13	07/05/13
03.03 - Implementació d'estadístiques i log	4 dies	08/05/13	11/05/13
03.04 - Elaboració jocs de proves (i correccions)	5 dies	13/05/13	17/05/13
03.05 - Elaboració de la documentació	2 dies	18/05/13	19/05/13
03.07 - Lliurament Documentació (PAC 3)	0 dies	19/05/13	19/05/13
04 - Documentació	21 dies	20/05/13	12/06/13
04.01 - Proves i depuració d'errors	10 dies	20/05/13	30/05/13
04.02 - Elaboració de la memòria	7 dies	31/05/13	07/06/13
04.03 - Elaboració de la presentació virtual	4 dies	08/06/13	12/06/13
04.04 - Lliurament Documentació (LLIURAMENT FINAL)	0 dies	12/06/13	12/06/13
05 - Debat Virtual	4 dies	25/06/13	28/06/13

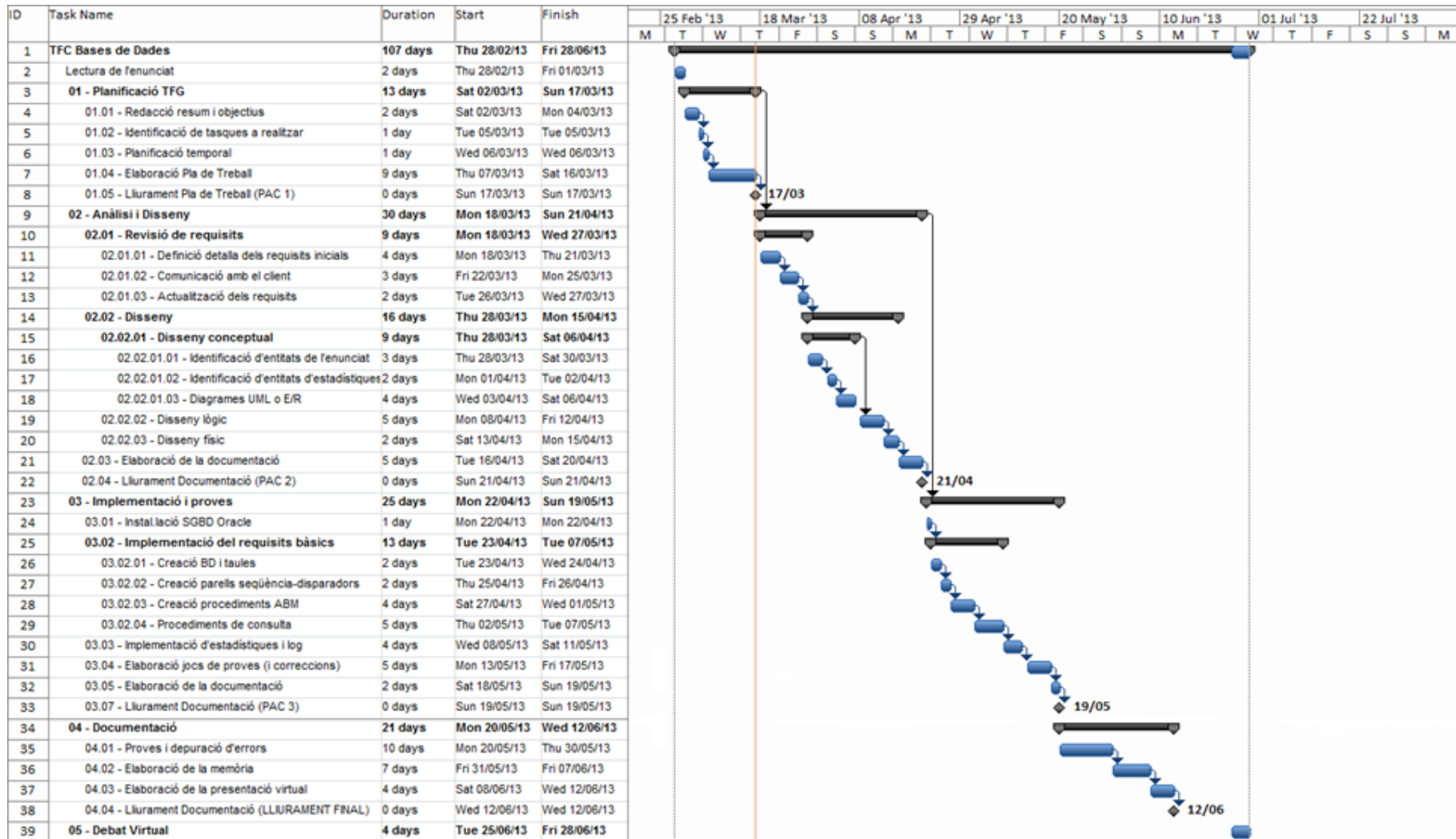
Observacions:

- La planificació és orientativa, a mesura que avanci el projecte és possible que hi hagi reajustaments del calendari tenint en compte els dies festius i les hores reals dedicades, a més, com hem comentat anteriorment, també hi poden haver imprevistos.
- La finalitat d'aquesta planificació és la d'orientar al client sobre la càrrega de treball que representa implantar el sistema requerit.

3.4.3 Diagrama de Gantt

Amb l'ajuda de Microsoft Project 2010 i considerant la data d'inici del projecte 28/02/2013, la planificació proposada és la següent:

Figura 1: Diagrama de Gantt



3.5 Productes obtinguts

3.5.1 Lliuraments

L'objectiu final del TFG és el lliurament dels elements següents:

- El **Pla de Treball** que recull la planificació temporal prevista de les tasques que componen la realització del total del projecte.
- El **producte final** o treball pràctic que inclou la BD, el codi SQL, etc.
- Una **memòria** que reculli tota la feina feta, d'un màxim de 90 planes.
- Una **presentació virtual** que resumeixi allò que s'ha fet, d'un màxim de 20 diapositives.

Cal destacar que el **pla de treball (PT) és molt important**, doncs determina com encararem el projecte i què s'entregarà en cada data. A més del lliurament final, hi ha diversos *lliuraments parcials* que cal anar fent i que corresponen a l'avaluació continuada de l'assignatura.

3.5.2 Documentació parcial

3.5.2.1 Planificació TFG (PAC 1: Pla de Treball)

Aquesta tasca s'haurà realitzat ja en realitzar el lliurament d'aquest Pla de Treball. Seguint les metodologies de planificació apreses, es determinaran les tasques necessàries per aconseguir els objectius del projecte i s'estimarà l'esforç i els recursos necessaris per dur-les a terme.

3.5.2.2 Anàlisi i Disseny (PAC 2)

Aquesta tasca comprèn les següents activitats:

- Una revisió dels requeriments proporcionats en l'enunciat per la BD, per avaluar si són suficients per començar el disseny i identificar els punts oberts que requereixin una clarificació.
- Elaboració del disseny conceptual de la BD mitjançant diagrames UML o E/R i la seva conseqüent translació a disseny lògic.
- Instal·lació i configuració del SGBD Oracle Express en els equips assignats al projecte, de manera que estigui disponible a l'inici del desenvolupament.
- Elaboració del disseny físic de la BD.

3.5.2.3 Implementació de la Base de Dades (PAC 3)

Aquesta tasca comprèn les següents activitats:

- Construcció i prova dels objectes de la BD incloent l'esquema de taules, joc de dades, seqüències i disparadors, els procediments emmagatzemats i les funcions necessàries.
- Revisió i proves del codi desenvolupat.

3.5.2.4 Elaboració Memòria i Presentació virtual (LLIURAMENT FINAL)

Aquesta tasca recull la realització i lliurament dels documents finals d'aquest TFG:

- El **producte final** o treball pràctic que inclou la BD, el codi SQL, etc.
- Una **memòria** que reculli tota la feina feta, d'un màxim de 90 planes.
- Una **presentació virtual** que resumeixi allò que s'ha fet, d'un màxim de 20 diapositives.

3.6 Breu descripció dels altres capítols de la memòria

Els capítols que completen la memòria són els següents:

4. Anàlisi de requisits: revisió dels requisits inicials i descripció detallada de les especificacions tècniques que ha de complir el producte.
5. Disseny: partint de les especificacions de l'anàlisi de requisits, en aquest capítol es realitza la fase de disseny conceptual, lògic i físic.
6. Implementació i proves: descripció de procediments i disparadors i creació dels scripts per a traduir a codi SQL la informació obtinguda a la fase de disseny. A més, aquest capítol inclou la creació d'un joc de proves per a garantir el correcte funcionament del sistema.
7. Recursos empleats i valoració econòmica: descripció dels recursos humans i materials utilitzats, juntament amb una estimació del cost del TFG.
8. Conclusions: conclusions obtingudes a la finalització del treball.
9. Glossari: glossari de termes específics vinculats al TFG.
10. Bibliografia: llistat de referències bibliogràfiques utilitzades.
11. Annexos: informació addicional sobre continguts de la memòria.

4 Anàlisi de requisits

En aquest capítol es fa l'anàlisi detallat dels requisits inicials especificats pel client i de les necessitats que haurà de resoldre el futur programari.

4.1 Descripció inicial

El nostre equip serà l'encarregat d'analitzar els requeriments del nou sistema, així com la posterior implementació de la base de dades per satisfer-los. Tota la gestió i accés a la informació es farà mitjançant procediments de base de dades, sent aquesta l'única manera d'accedir-hi. Per tant, no caldrà implementar cap interfície d'usuari amb un llenguatge de programació d'alt nivell.

Per realitzar el disseny es realitzarà primerament el model conceptual emprant el Diagrama E/R, incloent una llista amb totes les restriccions d'integritat. A nivell de procediments, caldrà implementar i descriure amb detall els procediments d'ABM (Alta + Baixa + Modificació) de totes les entitats que es considerin rellevants.

4.2 Requisits funcionals

Els requisits funcionals establerts per l'enunciat per al model de dades a dissenyar són els següents:

Taula 3: Regles de negoci per a la Base de Dades

Referència	Descripció
R01	<p>Els diferents productes que vol registrar l'empresa es qualifiquen tenint en compte les consideracions següents:</p> <ul style="list-style-type: none"> Hi ha un component bàsic que forma el producte i que es fabrica a partir d'una determinada fórmula. Cada producte es comercialitza en un determinat format que ve qualificat pel tipus d'envàs i per la quantitat que conté. Els productes es comercialitzen amb una determinada presentació (caixa, estoig, etc.) i sota una marca concreta. Per tal de poder vendre en un país determinat es necessita fer-ho a través d'un distribuïdor homologat en aquell país.
R02	<p>El registre d'un producte concret s'ha de fer per a tota combinació possible de marca, presentació, format i fórmula, i consisteix en presentar la descripció tècnica de tot aquest conjunt. Un cop enviada la documentació, les entitats sanitàries l'analitzen i poden donar alguna de les respostes següents:</p> <ul style="list-style-type: none"> Producte registrat: ja es pot començar a distribuir. Producte no registrat: l'empresa pot decidir llavors si recorre judicialment la decisió. Registre incomplet: en aquest cas l'empresa ha de respondre als diferents dubtes o requeriments que li facin les autoritats sanitàries del país en qüestió. <p>Un exemple de producte a registrar podria ser la colònia de nom "<i>Eau de pomme de terre</i>", de la marca "<i>Antoine Flags</i>", que té per fórmula de composició "$x4y5 + h7i9$", de 100 ml, que es comercialitza en envàs de vidre dins de <i>caixa metàl·lica</i>. Intentem registrar aquets producte a l'<i>Uruguai</i> a on tenim un distribuïdor homologat que es diu "<i>Distribuciones azules</i>".</p>

R03	Tota la gestió i accés a la informació es farà mitjançant procediments de base de dades, sent aquesta l'única manera d'accedir-hi. Per tant, no caldrà implementar cap interfície d'usuari amb un llenguatge de programació d'alt nivell.
R04	Per realitzar el disseny de la BD es realitzarà primerament el model conceptual emprant el Diagrama UML o E/R, incloent una llista amb totes les restriccions d'integritat que es considerin rellevants.
R05	A nivell de procediments, caldrà implementar i descriure amb detall els procediments d'ABM (Alta + Baixa + Modificació) de totes les entitats que es considerin rellevants.
R06	<p>També es vol definir un magatzem de dades (<i>data warehouse</i>) per extreure estadístiques de les dades emmagatzemades.</p> <p>La única restricció que ha de tenir aquest magatzem de dades és que ha d'oferir els diferents resultats que es defineixin en temps constant 1, és a dir, fent una SELECT sobre un registre d'una taula (que no sigui una vista calculada o materialitzada, ni fent servir funcions d'agregats: <i>sum</i>, <i>avg</i>, etc. amb <i>group by</i>) es podran definir les consultes estadístiques que es considerin més interessants per executar sobre el model descrit, però, com a mínim, s'hauran de poder obtenir els resultats següents:</p> <ol style="list-style-type: none"> 1. Els productes que costa més registrar (caldrà emmagatzemar el cost total de cada registre). 2. Els països a on els tràmits són més lents (considerant el temps entre l'inici dels tràmits de registre fins a la data de resposta final del país). 3. El temps mig de registre per producte i/o país (amb les mateixes consideracions que el cas anterior). 4. % de països que tenim a la BD que no tenen distribuïdor associat. 5. El número mig de registres per país. 6. % de formats de producte que mai s'han registrat. 7. Etc.
R07	La BD haurà de ser escalable per poder anar incorporant progressivament totes aquelles necessitats que sorgeixen durant la seva vigència.
R08	A part de presentar els scripts de creació de la BD, s'haurà de fer-ne la inicialització amb un conjunt de dades suficient i presentar un joc exhaustiu de proves que permeti garantir la correctesa de les funcionalitats implementades, així com el control d'errors i de situacions d'excepció.
R09	<p>Finalment, per tal de facilitar el manteniment del sistema, es valorarà molt positivament disposar de mecanismes que permetin resoldre potencials problemes d'integració amb la resta del sistema:</p> <ul style="list-style-type: none"> ▪ Un log de les accions fetes amb la Base de Dades. ▪ Mecanismes per testejar la funcionalitat de la Base de Dades. ▪ Etc.

Com a conseqüència de la revisió dels requisits funcionals, es van identificar una sèrie de punts oberts que van ser remesos al client (consultor) per a la seva contestació. Detallem la seva resolució en la taula següent.

Taula 4: Punts oberts dels requisits funcionals

Referència	Descripció	Resolució
PO01	A cada país, tenim un o podem tenir més d'un proveïdor homologat?	N'hi pot haver més d'un.
PO02	Els proveïdors són sempre per país o en podem tenir de globals (que actuïn a més d'un país)?	Poden actuar a més d'un país.
PO03	Els tipus d'envàs i presentació són lliures o hem de limitar-nos a alguns en concret?	Considera que són lliures ja que no entrarem en el procés d'homologació d'envasos.
PO04	Hem de diferenciar els països per continents o els podem considerar de forma global?	Com vulguis, no hi ha cap restricció en això ja que no demanem cap consulta per continent.
PO05	En el magatzem de dades, parla del cost total de cada registre. Es refereix a cost econòmic o temporal?	Es refereix a cost econòmic.
PO06	En relació a la pregunta anterior, registrar un producte té un cost econòmic?	Si, entre d'altres, el cost del personal dedicat, el cost dels tràmits al país de destí, etc.
PO07	Un dels punts demana "El número mig de registres per país", a que es refereix? El número mig respecte a dates? Altres registres? Productes?	Seria el número mig per any. No està ben identificat a l'enunciat.

Una vegada validada la nostra comprensió dels requeriments, procedim a realitzar el disseny de la Base de Dades.

4.3 Mòdul estadístic

Per a cadascuna de les sis consultes plantejades a l'enunciat (*requisit R06*), hem de crear una taula. Cadascuna d'aquestes taules s'alimenta dels procediments emmagatzemats de la base de dades i la consulta final es limita a una simple SELECT de temps constant 1, és a dir, que es llegeixen dades ja existents en aquestes taules.

4.4 Logs

Pel tema del *log* de les accions fetes, es recomana emmagatzemar totes les crides a procediments que es facin en una taula de logs, emmagatzemant el procediment executat, els paràmetres d'entrada i els de sortida. Per a estandaritzar el sistema que s'ha de fer, es demana explícitament que els procediments emmagatzemats compleixin les condicions següents:

- Com a mínim disposaran d'un paràmetre de sortida anomenat RSP, de tipus string, que indicarà si l'execució ha finalitzat amb èxit (valor 'OK') o si ha fracassat (valor 'ERROR+TIPUS D'ERROR').
- Disposaran de tractament d'excepcions.

5 Disseny

A la fase de disseny, s'especifica una solució als requeriments obtinguts a la fase anterior i se'n defineixen les estructures necessàries. Aquest disseny es divideix en tres etapes:

- **Disseny conceptual:** buscarem un disseny independent de la tecnologia que s'utilitzarà posteriorment. Obtindrem un model E/R com a resultat d'aquesta etapa.
- **Disseny lògic:** transformarem el model E/R en un model relacional ja que utilitzarem, com a tecnologia, un SGBD relacional.
- **Disseny físic:** buscarem optimitzar l'eficiència i completar el model amb alguns aspectes físics dependents del SGBD utilitzat (Oracle).

5.1 Disseny conceptual

A partir de l'anàlisi de requisits, en aquesta etapa s'obté una estructura de la informació de la futura BD independentment de la tecnologia que cal emprar. Per tant no es té en compte encara quin SGBD s'utilitzarà ni el llenguatge amb el que es farà la implementació.

El resultat del disseny conceptual s'expressarà utilitzant el model de dades anomenat *model entitat-relació*, que abreviarem amb model E/R.

5.1.1 Identificació d'entitats i atributs

5.1.1.1 Tipus d'entitats

Després de revisar els requisits exposats anteriorment, s'identifiquen tres tipus d'entitats:

- **Entitats principals:** són les entitats més importants del disseny que conformen la seva estructura bàsica. Totes les altres entitats completen a les entitats principals.
- **Entitats intermèdies:** són entitats que 'salven' una relació de molts-a-molts. Això és una mecànica coneguda del disseny de base de dades: quan dues entitats tenen una relació de molts-a-molts, hem de crear una entitat intermèdia que permeti crear una relació única (o no) entre les 2 entitats externes. Normalment, es crea una taula intermèdia amb una clau primària que compren els dos camps clau primària de les dues entitats externes.
- **Entitats auxiliars:** són entitats que necessitem per les dades que contenen. Podem vincular aquestes entitats auxiliars a les altres entitats o no fer-ho. La diferència és que si vinculem una entitat auxiliar a una altra entitat, l'atribut de l'entitat a la qual es vincula l'entitat auxiliar haurà de rebre un valor de l'entitat auxiliar de forma obligatòria. Si no ho fem i mantenim l'entitat auxiliar independent, haurem de gestionar l'atribut que ha de rebre el valor de forma manual, sense l'ajuda de la integritat referencial proporcionada pel SGBD. És important entendre aquesta diferència: la vinculació d'una entitat auxiliar a una altra entitat obliga a determinat atribut no solament a tenir un valor (és equivalent a NOT NULL) sinó a més al fet que aquest valor sigui un valor del domini de valors de l'entitat auxiliar.

Les entitats principals identificades són les següents:

- PRODUCTE
- PRESENTACIÓ
- MARCA
- PAÍS
- COST
- DISTRIBUÏDOR

Les entitats intermèdies identificades són:

- REGISTRE
- RESPOSTA
- FORMAT

Finalment, les entitats auxiliars identificades són totes les altres entitats que s'encarreguen d'emmagatzemar dades per a les entitats principals (com envàs, moneda i continent), les que constitueixen el *Mòdul Estadístic* i l'entitat que s'encarrega d'emmagatzemar els logs:

- ENVÀS
- MONEDA
- CONTINENT
- EST1_PROD_COST
- EST2_PAISOS_TRAM_LENT
- EST3_TEMPS_MIG
- EST4_PERC_DIST_PAISOS
- EST5_MIG_REG_PAIS
- EST6_FORMAT_NO_REG
- LOGS

5.1.1.2 Atributs de les entitats

Els atributs detectats per a les diferents entitats seran els següents (s'han subratllat els que componen les claus primàries):

PRODUCTE

idProducte, Nom, Formula, Format, Presentacio, Marca, Pais, Distribuidor, Registrat

- L'atribut *Formula* indica com està fabricat el producte.
- L'atribut *Pais* ens indica a on volem registrar el producte.
- L'atribut *Distribuidor* ens indica el distribuïdor amb el que treballarem a un país concret.
- L'atribut *Registrat* ens permet controlar si el producte es troba en estat REGISTRAT o NO REGISTRAT (No registrat, Incomplet).

FORMAT

idFormat, Envas, Quantitat

- L'atribut *Envas* indica el tipus d'envàs utilitzat.
- L'atribut *Quantitat* indica la quantitat de producte a cada envàs.

ENVAS

idEnvas, Tipus

- L'atribut *Tipus* indica la morfologia de l'envàs.

PRESENTACIO

idPresentacio, Tipus

- L'atribut *Tipus* indica el tipus de presentació (caixa, estoig, etc.).

MARCA

idMarca, Nom

- L'atribut *Nom* indica el nom de la marca.

REGISTRE

idRegistre, Resposta, Producte, País, Distribuidor, Cost, dataInici, dataFi, Durada

- L'atribut *Resposta* ens permet controlar l'estat del tràmit de registre.
- L'atribut *Producte* es refereix al producte que s'està registrant.
- L'atribut *País* ens indica a on volem registrar el producte.
- L'atribut *Distribuidor* ens indica el distribuïdor amb el que treballarem a un país concret.
- L'atribut *Cost* ens permet saber el cost del registre.
- Els atributs *dataInici*, *dataFi* i *Durada* permeten conèixer la durada dels tràmits de registre, els dos primers atributs estan en format YYYY-MM-DD HH24:MI:SS.

RESPOSTA

idResposta, Registrat, Estat

- L'atribut *Registrat* ens permet controlar l'estat del tràmit de registre.
- L'atribut *Estat* ens permet controlar si el tràmit de registre esta REGISTRAT o NO REGISTRAT (No registrat, Incomplet).

MONEDA

idMoneda, Nom, canviEuro

- L'atribut *canviEuro* ens indica el canvi en moneda estàndard (Euro - €).

COST

Cost, Producte, Quantitat, Moneda

- L'atribut *Cost* ens indica el canvi en moneda estàndard (Euro - €).
- L'atribut *Moneda* ens indica el tipus de moneda utilitzat.

PAIS

idPaís, Nom, Continent, Moneda

- L'atribut *Nom* indica el nom del país.
- L'atribut *Continent* indica el continent al que pertany el país.
- L'atribut *Moneda* indica la moneda pròpia de cada país.

CONTINENT

idContinent, Nom

- L'atribut *Nom* indica el nom del continent.

DISTRIBUIDOR

idDistribuidor, Nom, País, esHomologat

- L'atribut *esHomologat* indica si el distribuïdor està homologat al país.

LOGS

Codi, Data, Procediment, paramsEntrada, paramsSortida

- L'atribut *Data* indica la data en format YYYY-MM-DD HH24:MI:SS.
- Entitat per emmagatzemar la informació de log sobre l'execució de procediments emmagatzemats de la base de dades.

EST1_PROD_COST

Producte, Cost

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R06.1.

EST2_PAISOS_TRAM_LENT
Pais, duradaTramit

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R03.2.

EST3_TEMPS_MIG
Pais, mPais

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R06.3.

EST4_PERC_DIST_PAISOS
paisNoDist

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R06.4.

EST5_MIG_REG_PAIS
Pais, Anys, mRegistre

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R06.5.

EST6_FORMAT_NO_REG
FormatNoReg

- Entitat del mòdul estadístic que emmagatzema la informació necessària per a complir el requisit funcional R06.6.

5.1.1.3 Relació de les entitats

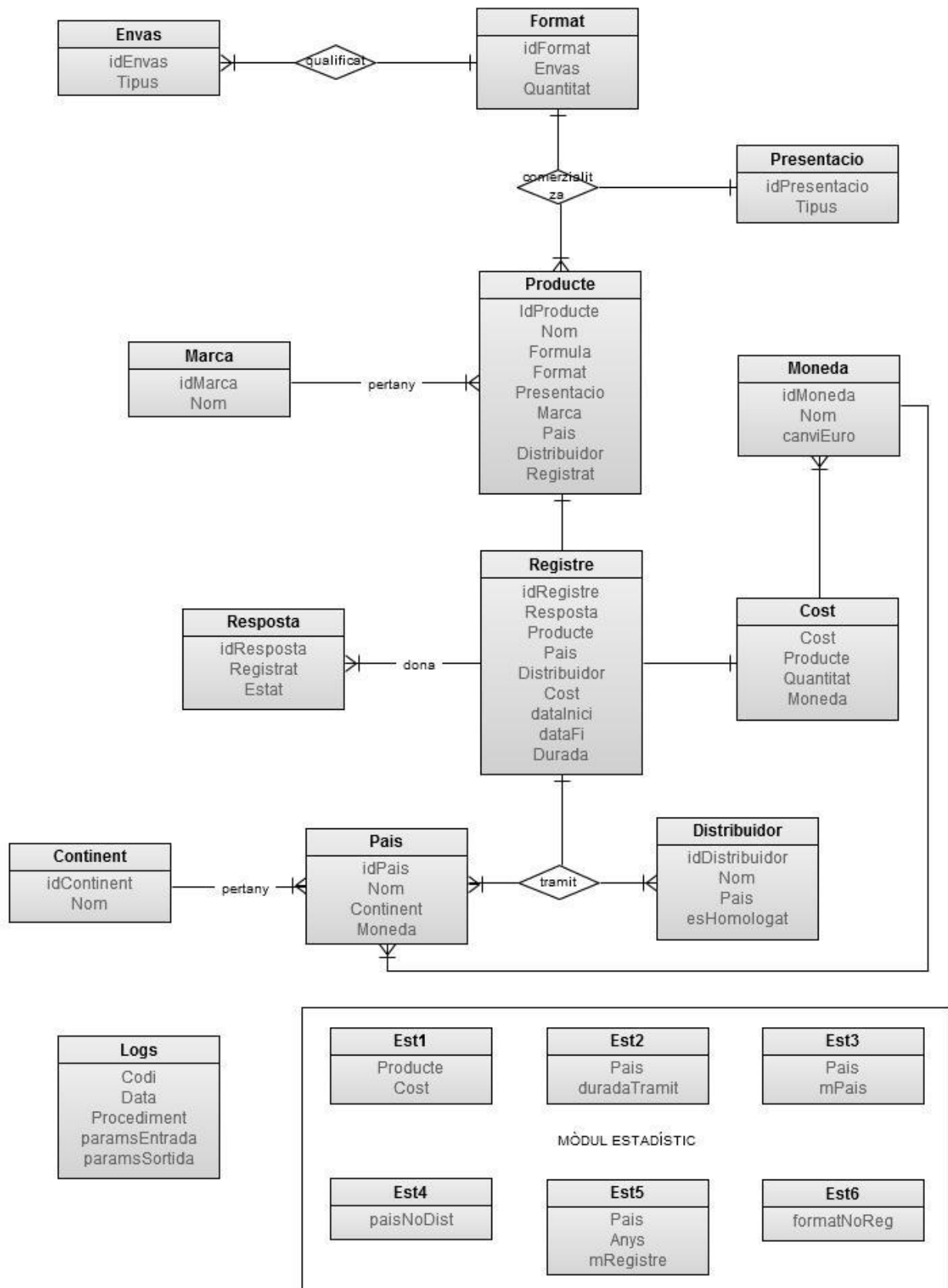
Entre les entitats descrites en el punt anterior s'estableixen les següents relacions:

Taula 5: Relacions entre entitats

Relació	Descripció	Connectivitat
Marca-Producte	Una marca pot tenir més d'un producte.	1:N
Presentacio-Format-Producte	Cada producte es comercialitza en un determinat format i amb una determinada presentació.	1:1:N
Format-Envas	Cada producte es comercialitza en un determinat format que ve qualificat pel tipus d'envàs i per la quantitat que conté.	1:N
Producte-Registre	Cada producte ha de passar un procés de registre.	1:1
Registre-Cost	Cada procés de registre té un cost concret.	1:1
Cost-Moneda	El cost pot venir determinat per diferents tipus de moneda.	1:N
Pais-Moneda	Hi ha una moneda pròpia de cada país.	1:1
Registre-Resposta	El registre d'un producte concret pot donar, inicialment, tres respostes diferents.	1:N
Registre-Pais-Distribuïdor	Per raons sanitàries, abans de poder començar a vendre cap producte cosmètic en un país <i>determinat</i> es necessita passar un conjunt de tràmits burocràtics que s'anomenen registre. Un cop registrat, cada distribuïdor pot vendre més d'un producte i, a la vegada, cada producte pot ser venut per més d'un distribuïdor.	1:N:M
Continent-Pais	A cada continent hi ha més d'un país.	1:N

5.1.2 Diagrama entitat/relació

Figura 2: Model E/R de la Base de Dades



5.1.3 Restriccions d'integritat

Durant la fase inicial d'anàlisi hem fet una recopilació en un llistat de les restriccions d'integritat que hauria de contemplar el disseny:

- Les taules *Resposta*, *Cost*, *Continent* i *País* només tenen procediment d'ALTA.
- Els distribuïdors poden actuar a més d'un país.
- Si un distribuïdor actua a més d'un país, tindrà un ID diferent a cada país.
- La moneda base en la que es calcularà el cost serà el Euro (€).
- Cada país té un sol tipus de moneda.
- Un cop iniciat el registre d'un producte, aquest ja no es podrà modificar. Si es vol modificar s'haurà de donar de baixa i tornar-lo a donar d'alta (amb un nou registre).
- A nivell estadístic, per a les dues primeres peticions (*productes que costa més registrar i països on els tràmits són més lents*), s'ha optat per fer un llistat de fins a 10 elements tot i que, en cas necessari, es podria modificar fàcilment.

5.2 Disseny lògic

En aquest etapa es transforma el resultat del disseny conceptual de manera que s'adapti a la tecnologia que cal emprar. Com que al nostre cas utilitzarem un SGBD relacional, el resultat d'aquesta etapa serà un conjunt de relacions amb els seus atributs, claus primàries i claus foranes. El resultat d'aquesta transformació és el *model relacional*. A partir del model E/R, es procedeix a la transformació en un model lògic relacional més afí, seguint les següents regles bàsiques:

- Les entitats originen relacions.
- Les interrelacions binàries 1:1 i 1:N originen claus foranes.
- Les interrelacions binàries M:N originen noves relacions.

5.2.1 Model relacional

PRODUCTE (idProducte, Nom, Formula, Format, Presentacio, Marca, Pais, Distribuidor, Registrat)

On {Marca} és clau forana a MARCA(idMarca)

On {Format} és clau forana a FORMAT(idFormat)

On {Presentacio} és clau forana a PRESENTACIÓ(idPresentacio)

On {Pais} és clau forana a PAIS(idPais)

On {Distribuidor} és clau forana a DISTRIBUIDOR(idDistribuidor)

On {Registrat} és clau forana a REGISTRE(idRegistre)

FORMAT (idFormat, Envas, Quantitat)

On {Envas} és clau forana a ENVAS(idEnvas)

ENVÀS (idEnvas, Tipus)

PRESENTACIÓ (idPresentacio, Tipus)

MARCA (idMarca, Nom)

REGISTRE (idRegistre, Resposta, Producte, Pais, Distribuidor, Cost, dataInici, dataFi, Durada)

On {Resposta} és clau forana a RESPOSTA(idResposta)

On {Producte} és clau forana a PRODUCTE(idProducte)

On {Pais} és clau forana a PAIS(idPais)

On {Distribuidor} és clau forana a DISTRIBUIDOR(idDistribuidor)

On {Cost} és clau forana a COST(cost)

RESPOSTA (idResposta, Registrat, Estat)

MONEDA (idMoneda, Nom, canviEuro)

COST (Cost, Producte, Quantitat, Moneda)

On {Producte} és clau forana a PRODUCTE(idProducte)

On {Moneda} és clau forana a MONEDA(idMoneda)

PAÍS (idPaís, Nom, Continent, Moneda)

On {Continent} és clau forana a CONTINENT(idContinent)

On {Moneda} és clau forana a Moneda(idMoneda)

CONTINENT (idContinent, Nom)

DISTRIBUÏDOR (idDistribuidor, Nom, País, esHomologat)

On {País} és clau forana a PAIS(idPaís)

LOGS (Codi, Data, Procediment, paramsEntrada, paramsSortida)

EST1_PROD_COST (Producte, Cost)

EST2_PAISOS_TRAM_LENT (País, duradaTramit)

EST3_TEMPS_MIG (País, mPaís)

EST4_PERC_DIST_PAISOS (paisNoDist)

EST5_MIG_REG_PAIS (País, Anys, mRegistre)

EST6_FORMAT_NO_REG (FormatNoReg)

5.3 Disseny físic

A partir del disseny lògic d'una base de dades hem de passar al disseny físic d'aquesta, passant pel nivell virtual. La forma d'implementar un disseny lògic en un SGBD concret depèn de les característiques pròpies de cada un d'aquests.

El primer que hem de fer és instal·lar el SGBD escollit, en el nostre cas Oracle Database 11g Express Edition, la versió exacta és *Oracle Database 11g Release 2 (11.2.0.1.0)*.

5.3.1 Instal·lació i configuració del SGBD Oracle

En paral·lel a la revisió de requeriments i el disseny, es va dur a terme la [instal·lació](#) del SGBD Oracle en la seva versió 11.2, que va comportar algunes dificultats.

Com que hem utilitzat una versió de 32-bit en una màquina de 64-bit, durant la instal·lació ens hem trobat amb el següent error:

```
The installer is unable to instantiate the file
C:\Users\Administrator\AppData\Local\Temp\{078E83D7-3FCC-4A72-903B-
995C7CE44681}\KEY_XE.reg. The file does not appear to exist.
```

L'arxiu físic no es troba, l'error és degut a que l'instal·lador no escrigui el valor d'una *clau del registre*. Per tant, es pot afegir el valor de la clau de registre a on falten dades i, a continuació, fer clic a *OK* al quadre de diàleg d'error per a continuar la instal·lació. La clau que falta es troba aquí, en el Registre de Windows:

```
HKEY_CLASSES_ROOT\Installer\Products\266B7A503A089BE4EAD1986A429434C1\SourceList\Media
```

Un cop ja s'ha instal·lat Oracle Database 11g Express Edition. Podem iniciar la línia d'ordres escrivint `sqlplus system/password` en una línia de comandes o anar a l'opció *Get Started* al menú de programes.

5.3.2 Creació de la Base de Dades

A Oracle, hi ha tres maneres de crear una base de dades:

- Utilitzar l'aplicació Database Configuration Assistant (DBCA).
DBCA pot ser llançat per Oracle Universal Installer, depenent del tipus d'instal·lació que es seleccioni, i proporciona una interfície gràfica d'usuari (GUI) que ens guiarà a través de la creació d'una Base de Dades.
També es pot iniciar DBCA com una eina independent, en qualsevol moment després de la instal·lació d'Oracle per crear o fer una còpia (clon) d'una Base de Dades.
- Utilitzar la sentència `CREATE DATABASE` per crear-la manualment.
Es pot utilitzar la instrucció `CREATE DATABASE` per crear una Base de Dades. En aquest cas, s'han de completar les accions addicionals abans de tenir una Base de Dades operacional. Podem consultar [Manually Creating an Oracle Database](#) per obtenir informació detallada sobre la creació d'una Base de Dades manualment.
- Actualitzar una Base de Dades existent.
Si ja existeix una versió anterior d'Oracle, només es requereix la creació d'una Base de Dades si se'n desitja una de nova. Es pot actualitzar la Base de Dades existent i utilitzar-la amb la nova versió del programari.

En aquest TFG assumirem que el client ja té creada i configurada una BD completament funcional i només ens faria falta crear el nou **espai per a taules** i l'**usuari** necessari.

5.3.2.1 Espai per a taules (tablespaces)

S'ha de crear el tablespace que actuaran com a unitat lògica d'emmagatzematge a dins la BD. Tots els elements que formen la Base de Dades (taules, índexs, etc.) es troben assignats a un tablespace.

Es diu que és una unitat lògica perquè no és visible des del sistema de arxius de l'ordinador on es troba la BD. Al mateix temps un tablespace té, com a mínim, un fitxer de dades associat que es troba físicament al sistema d'arxius de l'ordinador. Per tant, el tablespace fa de pont entre la BD i el disc físic.

Per raons d'eficiència convé separar les taules dels índexs en diferents tablespaces. Per tant crearem dos tablespaces. Com que inicialment no es coneix el volum de les dades que tindrà la BD, es crearan amb una grandària inicial de 50 Mb, que s'anirà auto estenent en cas de necessitar més espai. L'avaluació de l'espai necessari la farà en la BD, ja que s'utilitza la opció 'AUTOALLOCATE'.

S'ha de tenir en compte que, en aquest script, ve indicada la ruta que crea Oracle Express per defecte: `'C:\oracle\app\oracle\oradata\XE'`. El script de creació d'aquests tablespaces es troba a `0-Tablespaces.sql` que s'adjunta amb el document.

```
-- Connectat com SYSTEM

-- Tablespace per taules
CREATE TABLESPACE TFG_DADES
DATAFILE 'C:\oracle\app\oracle\oradata\XE\tfg_dades.dbf' SIZE 50M
AUTOEXTEND ON
LOGGING
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;
```



```
-- Tablespace per índexs
CREATE TABLESPACE TFG_INDEX
DATAFILE 'C:\oracle\app\oracle\oradata\XE\tfg_indexs.dbf' SIZE 50M
AUTOEXTEND ON
LOGGING
PERMANENT
EXTENT MANAGEMENT LOCAL AUTOALLOCATE;

COMMIT;
```

5.3.2.2 Usuaris

Es crea l'usuari 'JAUME' amb el qual es podrà accedir a les diferents taules i funcionalitats de la BD. Aquest usuari serà propietari de tots els objectes (taules, índexs, procedures, etc.) de la mateixa. Ha de tenir els privilegis suficients per poder crear els objectes necessaris i accedir als tablespaces TFG_DADES i TFG_INDEX.

L'usuari, a més, a de tenir els rols CONNECT i RESOURCE assignats i quota il·limitada sobre els tablespaces esmentats anteriorment.

```
-- Connectat com SYSTEM

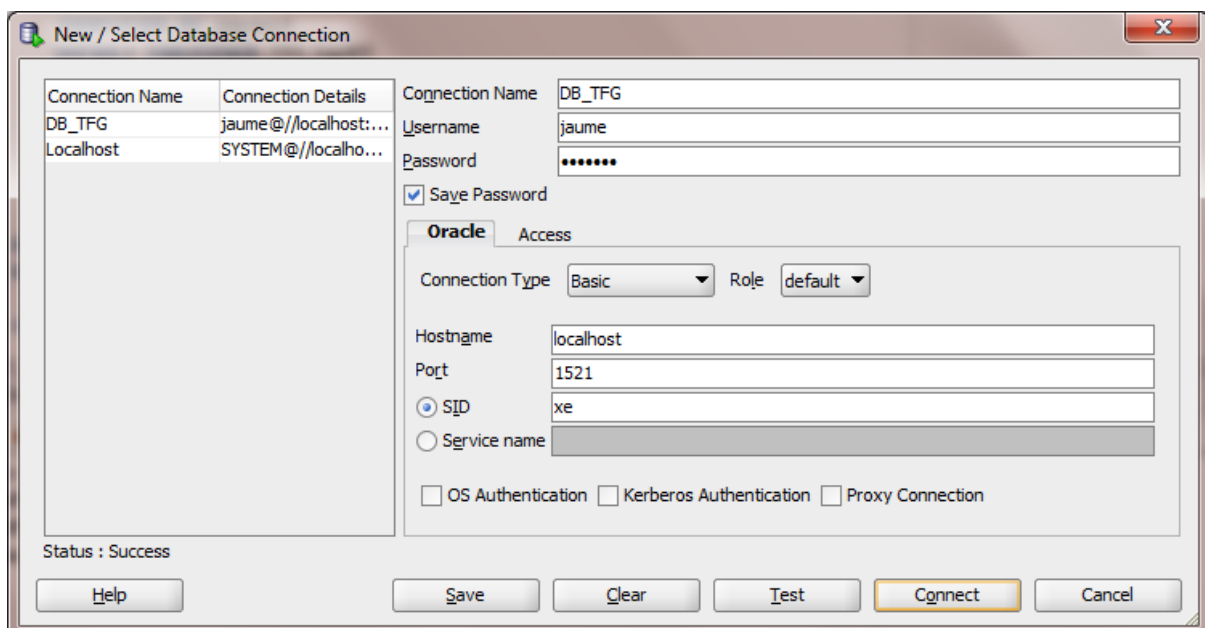
CREATE USER JAUME
IDENTIFIED BY Gandalf1714
DEFAULT TABLESPACE TFG_DADES
QUOTA UNLIMITED ON TFG_DADES
QUOTA UNLIMITED ON TFG_INDEX

TEMPORARY TABLESPACE TEMP
PROFILE DEFAULT
ACCOUNT UNLOCK;

GRANT CONNECT, RESOURCE TO JAUME;

COMMIT;
```

Un cop creat l'usuari farem una connexió a la BD via *Oracle SQL Developer*:



El script de creació de l'usuari es troba a 1-CreacioUsuari que s'adjunta amb el document.

5.3.3 Creació de taules

A partir de la informació del model relacional, es defineixen les taules que s'han de crear a la Base de Dades. S'indica el tipus d'atribut utilitzat especificant els que són obligatoris (NOT NULL) i es consideren essencials pel correcte funcionament del sistema.

També es descriuen altres restriccions sobre alguns atributs, cal destacar la utilització de clàusules CHECK en alguns camps per assegurar-nos que el seu contingut s'ajustarà als requisits.

5.3.3.1 Taules del mòdul principal

A continuació es detallen les taules creades a la Base de Dades. El script de creació d'aquestes taules es troba a 2-CreacioTaules.sql que s'adjunta amb el document.

Taula			
MARCA			
Camps	Tipus	Nul	Descripció
IDMARCA	NUMBER	NOT NULL	Identificador de marca
NOM	VARCHAR2(20)	NOT NULL	Nom de la marca
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_MARCA	Primary key		IDMARCA
UN_MARCA	Unique index		NOM

Taula			
ENVAS			
Camps	Tipus	Nul	Descripció
IDENVAS	NUMBER	NOT NULL	Identificador d'envàs
TIPUS	VARCHAR2(20)	NOT NULL	Tipus d'envàs
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_ENVAS	Primary key		IDENVAS
UN_ENVAS	Unique index		TIPUS

Taula			
FORMAT			
Camps	Tipus	Nul	Descripció
IDFORMAT	NUMBER	NOT NULL	Identificador del format
ENVAS	NUMBER	NOT NULL	Identificador d'envàs
QUANTITAT	VARCHAR2(8)	NOT NULL	Quantitat de producte
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_FORMAT	Primary key		IDFORMAT
UN_FORMAT	Unique index		ENVAS, QUANTITAT
FK_FORMAT_ENVAS	Foreign key	ENVAS	IDENVAS

Taula			
PRESENTACIO			
Camps	Tipus	Nul	Descripció
IDPRESENTACIO	NUMBER	NOT NULL	Identificador de la presentació
TIPUS	VARCHAR2(50)	NOT NULL	Tipus de presentació
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_PRESENTACIO	Primary key		IDPRESENTACIO
UN_PRESENTACIO	Unique index		TIPUS

Taula			
RESPOSTA			
Camps	Tipus	Nul	Descripció
IDRESPOSTA	NUMBER	NOT NULL	Identificador de la resposta
REGISTRAT	CHAR(2)	NOT NULL	Resposta del registre
ESTAT	VARCHAR2(20)	NOT NULL	Estat del registre

Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_RESPOSTA	Primary key		IDRESPOSTA
UN_RESPOSTA	Unique index		REGISTRAT, ESTAT
CK_RESPOSTA	Check		Valors possibles: SI + REGISTRAT NO + NO REGISTRAT NO + INCOMPLET

Taula			
MONEDA			
Camps	Tipus	Nul	Descripció
IDMONEDA	NUMBER	NOT NULL	Identificador de la moneda
NOM	VARCHAR2(20)	NOT NULL	Nom de la moneda
CANVIEURO	NUMBER		Canvi de euro (€) a la moneda local
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_MONEDA	Primary key		IDMONEDA
UN_MONEDA	Unique index		NOM
CK_MONEDA	Check		CANVIEURO > 0

Taula			
CONTINENT			
Camps	Tipus	Nul	Descripció
IDCONTINENT	NUMBER	NOT NULL	Identificador del continent
NOM	VARCHAR2(50)	NOT NULL	Nom del continent
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_CONTINENT	Primary key		IDCONTINENT
UN_CONTINENT	Unique index		NOM

Taula			
PAIS			
Camps	Tipus	Nul	Descripció
IDPAIS	NUMBER	NOT NULL	Identificador del país
NOM	VARCHAR2(50)	NOT NULL	Nom del país
CONTINENT	NUMBER	NOT NULL	Identificador del continent
MONEDA	NUMBER	NOT NULL	Identificador de la moneda
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_PAIS	Primary key		IDPAIS
UN_PAIS	Unique index		NOM
FK_PAIS_CONTINENT	Foreign key	CONTINENT	IDCONTINENT
FK_PAIS_MONEDA	Foreign key	MONEDA	IDMONEDA

Taula			
DISTRIBUIDOR			
Camps	Tipus	Nul	Descripció
IDDISTRIBUIDOR	NUMBER	NOT NULL	Identificador del distribuïdor
NOM	VARCHAR2(100)	NOT NULL	Nom del distribuïdor
PAIS	NUMBER	NOT NULL	Identificador del país
ESHOMOLOGAT	CHAR(2)	NOT NULL	Indica si el distribuïdor està homologat al país
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_DISTRIBUIDOR	Primary key		IDDISTRIBUIDOR
UN_DISTRIBUIDOR	Unique index		NOM, PAIS
FK_DIST_PAIS	Foreign key	PAIS	IDPAIS
CK_DISTRIBUIDOR	Check		Valors possibles: SI/NO

Taula			
PRODUCTE			
Camps	Tipus	Nul	Descripció
IDPRODUCTE	NUMBER	NOT NULL	Identificador del producte
NOM	VARCHAR2 (50)	NOT NULL	Nom del producte
FORMULA	VARCHAR2 (20)	NOT NULL	Fórmula de fabricació
MARCA	NUMBER	NOT NULL	Identificador de la marca
FORMAT	NUMBER	NOT NULL	Identificador del format
PRESENTACIO	NUMBER	NOT NULL	Identificador de la presentació
PAIS	NUMBER	NOT NULL	Identificador del país
DISTRIBUIDOR	NUMBER	NOT NULL	Identificador del distribuïdor
REGISTRAT	NUMBER	NOT NULL	Estat del registre
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_PRODUCTE	Primary key		IDPRODUCTE
UN_PRODUCTE	Unique index		NOM, FORMULA
FK_PROD_MARCA	Foreign key	MARCA	IDMARCA
FK_PROD_FORMAT	Foreign key	FORMAT	IDFORMAT
FK_PROD_PRESENTACIO	Foreign key	PRESENTACIO	IDPRESENTACIO
FK_PROD_PAIS	Foreign key	PAIS	IDPAIS
FK_PROD_DIST	Foreign key	DISTRIBUIDOR	IDDISTRIBUIDOR
FK_PROD_REGISTRE	Foreign key	RESPOSTA	IDRESPOSTA

Taula			
COST			
Camps	Tipus	Nul	Descripció
COST	NUMBER	NOT NULL	Cost del registre (€)
PRODUCTE	NUMBER	NOT NULL	Identificador del producte
QUANTITAT	NUMBER	NOT NULL	Cost del registre en moneda local
MONEDA	NUMBER	NOT NULL	Nom de la moneda
Constraint	Tipus	Taula referenciada	Camps/Condicció
FK_COST_PRODUCTE	Foreign key	PRODUCTE	IDPRODUCTE
FK_COST_MONEDA	Foreign key	MONEDA	IDMONEDA
CK_COST	Check		QUANTITAT > 0

Taula			
REGISTRE			
Camps	Tipus	Nul	Descripció
IDREGISTRE	NUMBER	NOT NULL	Identificador del registre
RESPOSTA	NUMBER	NOT NULL	Estat del tràmit
PRODUCTE	NUMBER		Identificador del producte
PAIS	NUMBER	NOT NULL	Identificador del país
DISTRIBUIDOR	NUMBER	NOT NULL	Identificador del distribuïdor
COST	NUMBER		Cost del registre
DATAINICI	DATE	NOT NULL	Data inici del registre
DATAFI	DATE		Data fi del registre
DURADA	NUMBER		Durada del registre
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_REGISTRE	Primary key		IDREGISTRE
FK_REG_RESP	Foreign key	RESPOSTA	IDRESPOSTA
FK_REG_PROD	Foreign key	PRODUCTE	IDPRODUCTE
FK_REG_PAIS	Foreign key	PAIS	IDPAIS
FK_REG_DIST	Foreign key	DISTRIBUIDOR	IDDISTRIBUIDOR
CK_REG_COST	Check		COST > 0
CK_REG_DATA	Check		DATAINICI >= DATAFI
CK_REG_DURADA	Check		DURADA >= 0

5.3.3.2 Taules del mòdul estadístic

El **mòdul d'estadístic** té taules que, segons els requisits funcionals, han d'oferir els diferents resultats definits en temps constant 1, és a dir, fent una `SELECT` sobre un registre d'una taula (que no sigui una vista calculada, ni fent servir funcions d'agregats amb `group by`).

A continuació es detallen les taules creades a la Base de Dades. El script de creació d'aquestes taules es troba a `2-CreacioTaules.sql` que s'adjunta amb el document.

Taula			
EST1_PROD_COST			
Camps	Tipus	Nul	Descripció
PRODUCTE	NUMBER	NOT NULL	Identificador del producte
COST	NUMBER	NOT NULL	Cost del registre en euros (€)
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST1	Primary key		PRODUCTE

Taula			
EST2_PAISOS_TRAM_LENT			
Camps	Tipus	Nul	Descripció
PAIS	NUMBER	NOT NULL	Identificador del país
DURADATRAMIT	NUMBER	NOT NULL	Duració del tràmit
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST2	Primary key		PAIS

Taula			
EST3_TEMPS_MIG			
Camps	Tipus	Nul	Descripció
PAIS	NUMBER	NOT NULL	Identificador del país
MPAIS	NUMBER	NOT NULL	Temps mig de registre per país
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST3	Primary key		PAIS, MPAIS
FK_EST3_PAIS	Foreign key	PAIS	IDPAIS

Taula			
EST4_PERC_DIST_PAISOS			
Camps	Tipus	Nul	Descripció
PAISNODIST	NUMBER (5)	NOT NULL	% de països a la BD sense distribuïdor
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST4	Primary key		PAISNODIST

Taula			
EST5_MIG_REG_PAIS			
Camps	Tipus	Nul	Descripció
PAIS	NUMBER	NOT NULL	Identificador del país
ANYS	NUMBER	NOT NULL	Any concret demanat
MREGISTRE	NUMBER	NOT NULL	Número mig de registres per país
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST5	Primary key		PAIS

Taula			
EST6_FORMAT_NO_REG			
Camps	Tipus	Nul	Descripció
FORMATNOREG	NUMBER	NOT NULL	% de format no registrat
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_EST6	Primary key		FORMATNOREG

Les crides als procediments emmagatzemats del sistema s'emmagatzemaran en una **taula de logs**, la qual tindrà informació sobre els paràmetres d'entrada i sortida dels mateixos.

La taula de logs tindrà els següents camps:

LOGS (Codi, Data, Procediment, paramsEntrada, paramsSortida)

A on:

- L'atribut *Data* serà la data en la qual s'insereix aquesta entrada
- L'atribut *Procediment* serà el nom del procediment emmagatzemat.
- L'atribut *paramsEntrada* tindrà els paràmetres d'entrada, separats per comes.
- L'atribut *paramsSortida* tindrà els paràmetres de sortida igualment separats per comes.

A continuació es detalla la taula creada a la Base de Dades. El script de creació d'aquestes taules es troba a `2-CreacioTaules.sql` que s'adjunta amb el document.

Taula			
LOGS			
Camps	Tipus	Nul	Descripció
CODI	NUMBER	NOT NULL	Identificador del Log
DATA	DATE	NOT NULL	Data del Log
PROCEDIMENT	VARCHAR2 (50)	NOT NULL	Procediment executat
PARAMSENTRADA	VARCHAR2 (200)	NOT NULL	Paràmetres d'Entrada separats per comes
PARAMSORTIDA	VARCHAR2 (200)	NOT NULL	Paràmetres de Sortida separats per comes
Constraint	Tipus	Taula referenciada	Camps/Condicció
PK_LOGS	Primary key		CODI

5.3.4 Índexs

Els índexs són uns elements del disseny físic de la base de dades que tenen com a finalitat millorar el rendiment de les aplicacions quan accedeixen a les taules.

Oracle crea automàticament un índex de tipus únic i no nul (`UNIQUE`, `NOT NULL`) per cada clau primària que es declara a la Base de Dades. També crea automàticament el mateix tipus d'índex per a cada clau alternativa (`UNIQUE`) que es declari. Durant la creació de les taules s'ha especificat que aquests índexs s'han de guardar al tablespace que s'ha creat amb aquest objectiu (`TFG_INDEX`).

Seguint les recomanacions d'Oracle (*best-practices*) es crea un índex per cada clau forana, d'aquesta forma s'eviten problemes de rendiment, concurrències i bloquejos. El script de creació d'aquests índexs es troba a `3-Index.sql` que s'adjunta amb el document.

En principi no es considera crear més índexs addicionals a aquests, ja que el volum de dades i la freqüència d'ús del sistema no es preveu que sigui massa elevada, amb el que en agregar més índexs s'augmentaria la sobrecàrrega (ja que els índexs d'una taula es creen de nou amb cada operació *insert*, *update* o *delete* sobre ella) sense millorar sensiblement el rendiment.

6 Implementació i proves

En aquest capítol es descriu la implementació dels elements (procediments, seqüències i disparadors) de la Base de Dades necessaris per a realitzar les funcionalitats demanades als requisits.

6.1 Parells seqüència – disparador

A la fase de disseny s'ha decidit que la majoria de les taules tindran un camp que fa de codi identificador únic, que serà assignat automàticament per la Base de Dades en el moment de la inserció de nous registres.

Per a implementar aquest comportament es defineixen parells seqüència – disparador associats a les taules de la BD. Cada vegada que s'introdueix un nou registre a la BD, el disparador associat inserta el codi identificador corresponent a partir de la seqüència corresponent. D'aquesta forma s'assegura que no s'introdueixin dades a cap taula amb codi identificador repetit.

Excepcionalment, la taula *Cost* no inclou aquest funcionament automàtic, ja en aquesta no hi ha un codi identificador únic. Tampoc s'inclou aquest codi a les taules estadístiques ja que no s'introdueixen les dades manualment i no és necessari per a cap funcionalitat.

El script de creació d'aquests parells seqüència – disparador es troba a `4-Triggers.sql` que s'adjunta amb el document.

6.2 Implementació de funcionalitats

L'especificació dels tots els procediments inclou les descripcions del seu propòsit, paràmetres d'entrada, paràmetres de sortida i les verificacions que es duen a terme durant l'execució.

Els paràmetres de sortida inclouen com a mínim un paràmetre de tipus *string* anomenat RSP, que indica si l'execució a finalitzat correctament (valor 'OK') o si ha fracassat (valor 'ERROR+TIPIUS ERROR'). Per a tots els casos, el valor d' RSP serà introduït a la taula LOGS mostrant el resultat de l'execució.

Tant els procediments ABM com les consultes poden ser utilitzats per l'usuari de la BD, mentre que els procediments del mòdul estadístic i logs són d'ús intern, i s'executen automàticament quan s'utilitzen altres procediments.

6.2.1 Procediments d'alta, baixa i modificació (ABM)

A nivell de procediments, cal implementar i descriure amb detall els procediments d'ABM (Alta + Baixa + Modificació) de totes les entitats que són rellevants.

SP_ALTA_MARCA
Propòsit
- Afegeix una nova marca a la taula MARCA.
Paràmetres d'entrada
- p_nom VARCHAR2(20): nom de la marca.
Paràmetres de sortida
- RSP VARCHAR2
Verificacions
- Verifica que els camps obligatoris no siguin nuls.
- Verifica que la marca no existeixi prèviament a la taula.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: ja existeix una marca amb el mateix: nom' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_MARCA

Propòsit

- Elimina una marca de la taula MARCA.

Paràmetres d'entrada

- p_idMarca NUMBER: identificador de la marca.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que la marca existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: la marca no existeix'
'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_MODIFICACIO_MARCA

Propòsit

- Modifica dades d'una marca a la taula MARCA.

Paràmetres d'entrada

- p_idMarca NUMBER: identificador de la marca.
- p_nom VARCHAR2(20): nom de la marca.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que la marca existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: la marca no existeix'
'ERROR: ja existeix una marca amb el mateix: nom'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_ENVAS

Propòsit

- Afegeix un nou envàs a la taula ENVAS.

Paràmetres d'entrada

- p_tipus VARCHAR2(20): tipus de l'envàs.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que l'envàs no existeixi prèviament a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: ja existeix un envàs amb el mateix: tipus'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_ENVAS

Propòsit

- Elimina un envàs de la taula ENVAS.

Paràmetres d'entrada

- p_idEnvas NUMBER: identificador de l'envàs.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que l'envàs existeixi a la taula.

Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: l'envàs no existeix' 'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_ENVAS
Propòsit - Modifica dades d'un envàs a la taula ENVAS.
Paràmetres d'entrada - p_idEnvas NUMBER: identificador de l'envàs. - p_tipus VARCHAR2(20): tipus de l'envàs.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que l'envàs existeixi a la taula.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: l'envàs no existeix' 'ERROR: ja existeix un envàs amb el mateix: tipus' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_FORMAT
Propòsit - Afegeix un nou format a la taula FORMAT.
Paràmetres d'entrada - p_envas NUMBER: tipus d'envàs. - p_quantitat VARCHAR2(8): quantitat de producte.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que l'envàs existeix.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: l'envàs no existeix' 'ERROR: ja existeix un envàs amb el mateix: envas' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_BAIXA_FORMAT
Propòsit - Elimina un format de la taula FORMAT.
Paràmetres d'entrada - p_idFormat NUMBER: identificador del format.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el format existeixi a la taula.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el format no existeix' 'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_MODIFICACIO_FORMAT

Propòsit

- Modifica dades d'un format a la taula FORMAT.

Paràmetres d'entrada

- p_idFormat NUMBER: identificador del format
 - p_envas NUMBER: tipus d'envàs.
 - p_quantitat VARCHAR2(8): quantitat de producte.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
 - Verifica que el format existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
 'OK'
 - si l'execució ha fracassat retorna un dels següents missatges:
 'ERROR: el camp {camp} no pot ser nul'
 'ERROR: el format no existeix'
 'ERROR: ja existeix un format amb el mateix: envàs, quantitat'
 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_PRESENTACIO

Propòsit

- Afegeix una nova presentació a la taula PRESENTACIO.

Paràmetres d'entrada

- p_tipus VARCHAR2(20): tipus de presentació.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
 - Verifica que la presentació no existeixi prèviament a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
 'OK'
 - si l'execució ha fracassat retorna un dels següents missatges:
 'ERROR: el camp {camp} no pot ser nul'
 'ERROR: ja existeix una presentació amb el mateix: tipus'
 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_PRESENTACIO

Propòsit

- Elimina una presentació de la taula PRESENTACIO.

Paràmetres d'entrada

- p_idEnvas NUMBER: identificador de la presentació.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
 - Verifica que la presentació existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
 'OK'
 - si l'execució ha fracassat retorna un dels següents missatges:
 'ERROR: el camp {camp} no pot ser nul'
 'ERROR: la presentació no existeix'
 'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.'
 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_MODIFICACIO_PRESENTACIO

Propòsit

- Modifica dades d'una presentació a la taula PRESENTACIO.

Paràmetres d'entrada

- p_idEnvas NUMBER: identificador de la presentació.
 - p_tipus VARCHAR2(20): tipus de presentació.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions
<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que la presentació existeixi a la taula.
Retorna
<ul style="list-style-type: none"> - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: la presentació no existeix' 'ERROR: ja existeix una presentació amb el mateix: tipus' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_RESPOSTA	
Propòsit	- Afegeix una nova resposta a la taula RESPOSTA.
Paràmetres d'entrada	<ul style="list-style-type: none"> - p_registrat CHAR(2): estat del registre (Si/No). - p estat VARCHAR2(20): resposta del registre.
Paràmetres de sortida	- RSP VARCHAR2
Verificacions	<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que la resposta no existeixi prèviament a la taula. - Verifica que el valor del camp registrat sigui correcte. - Verifica que el valor del camp estat sigui correcte.
Retorna	<ul style="list-style-type: none"> - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: ja existeix una resposta amb el mateix: registrat, estat' 'ERROR: el valor dels camps és incorrecte' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_MONEDA	
Propòsit	- Afegeix una nova moneda a la taula MONEDA.
Paràmetres d'entrada	<ul style="list-style-type: none"> - p_nom VARCHAR2(20): nom de la moneda. - p_canviEuro NUMBER: canvi d'euro (€) a la moneda local.
Paràmetres de sortida	- RSP VARCHAR2
Verificacions	<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que la moneda no existeixi prèviament a la taula. - Verifica que el valor del camp canviEuro sigui correcte.
Retorna	<ul style="list-style-type: none"> - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: ja existeix una moneda amb el mateix: nom' 'ERROR: el valor del camp canviEuro ha de ser positiu' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_MONEDA	
Propòsit	- Elimina una moneda a la taula MONEDA.
Paràmetres d'entrada	- p idMoneda NUMBER: identificador de moneda.
Paràmetres de sortida	- RSP VARCHAR2
Verificacions	<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que la moneda existeixi a la taula.

Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: la moneda no existeix' 'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_MONEDA
Propòsit - Modifica dades d'una moneda a la taula MONEDA.
Paràmetres d'entrada - p_idMoneda NUMBER: identificador de la moneda. - p_nom VARCHAR2(20): nom de la moneda. - p_canviEuro NUMBER: canvi de euro (€) a la moneda local.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que la moneda existeixi a la taula.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: la moneda no existeix' 'ERROR: ja existeix una moneda amb el mateix: nom' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_COST
Propòsit - Afegeix un nou cost a la taula COST calculant el canvi d'euro (€) a moneda local. Es realitza per a cada producte registrat.
Paràmetres d'entrada - p_producte NUMBER: identificador del producte. - p_quantitat NUMBER: cost del registre en euro (€). - p_moneda VARCHAR2(20): nom de la moneda.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el producte existeix. - Verifica que la moneda existeix.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: {camp} no existeix' 'ERROR: el valor del camp quantitat ha de ser positiu' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_CONTINENT
Propòsit - Afegeix un nou continent a la taula CONTINENT.
Paràmetres d'entrada - p_nom VARCHAR2(20): nom del continent.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el continent no existeixi prèviament a la taula.

Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: ja existeix un continent amb el mateix: nom' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_PAIS
Propòsit - Afegeix un nou país a la taula PAIS.
Paràmetres d'entrada - p_nom VARCHAR2(20): nom del país. - p_continent NUMBER: identificador del continent. - p_moneda NUMBER: identificador de la moneda.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el país no existeixi prèviament a la taula. - Verifica que el continent existeix. - Verifica que la moneda existeix.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: {camp} no existeix' 'ERROR: ja existeix un país amb el mateix: nom' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_DISTRIBUIDOR
Propòsit - Afegeix un nou distribuïdor a la taula DISTRIBUIDOR.
Paràmetres d'entrada - p_nom VARCHAR2(100): nom del distribuïdor. - p_pais NUMBER: identificador del país. - p_esHomologat CHAR(2): indica si el distribuïdor esta homologat al país.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el distribuïdor no existeixi prèviament a la taula. - Verifica que el país existeix.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el país no existeix' 'ERROR: ja existeix un distribuïdor amb el mateix: nom, país' 'ERROR: el valor del camp esHomologat ha de ser SI/NO' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_DISTRIBUIDOR
Propòsit - Elimina un distribuïdor a la taula DISTRIBUIDOR.
Paràmetres d'entrada - p_idDistribuïdor NUMBER: identificador del distribuïdor.
Paràmetres de sortida - RSP VARCHAR2
Verificacions - Verifica que els camps obligatoris no siguin nuls. - Verifica que el distribuïdor existeixi a la taula.

Retorna
- si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el distribuïdor no existeix' 'ERROR: no es pot esborrar perquè està relacionada amb algun producte existent.' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_DISTRIBUIDOR
Propòsit
- Modifica dades d'un distribuïdor a la taula DISTRIBUIDOR.
Paràmetres d'entrada
- p_idDistribuïdor NUMBER: identificador del distribuïdor. - p_nom VARCHAR2(100): nom del distribuïdor. - p_pais NUMBER: identificador del país. - p_esHomologat CHAR(2): indica si el distribuïdor esta homologat al país.
Paràmetres de sortida
- RSP VARCHAR2
Verificacions
- Verifica que els camps obligatoris no siguin nuls. - Verifica que el distribuïdor existeixi a la taula.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el distribuïdor no existeix' 'ERROR: el valor del camp esHomologat ha de ser SI/NO' 'ERROR: ja existeix un distribuïdor amb el mateix: nom, país' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_PRODUCTE
Propòsit
- Afegeix un nou producte a la taula PRODUCTE. - L'alta del producte també serveix per iniciar els tràmits del registre.
Paràmetres d'entrada
- p_nom VARCHAR2(50): nom del producte. - p_formula VARCHAR2(20): fórmula de fabricació. - p_marca NUMBER: codi de la marca. - p_format NUMBER: codi del format. - p_presentacio NUMBER: codi de la presentació. - p_pais NUMBER: identificador del país. - p_distribuïdor NUMBER: identificador del distribuïdor.
Paràmetres de sortida
- RSP VARCHAR2
Verificacions
- Verifica que els camps obligatoris no siguin nuls. - Verifica que el producte no existeixi prèviament a la taula. - Verifica que el format existeix. - Verifica que la presentació existeix. - Verifica que la marca existeix. - Verifica que el país existeix. - Verifica que el distribuïdor existeix. - Verifica que el distribuïdor és del país corresponent.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: {camp} no existeix' 'ERROR: el distribuïdor no pertany al país' 'ERROR: ja existeix un producte amb el mateix: nom, fórmula' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_PRODUCTE

Propòsit

- Elimina un producte a la taula PRODUCTE.

Paràmetres d'entrada

- p_idProducte NUMBER: identificador del producte.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que el producte existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: el producte no existeix'
'ERROR: no es pot esborrar el producte perquè esta en procés de registre.'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_MODIFICACIO_PRODUCTE

Propòsit

- Modifica dades d'un producte a la taula PRODUCTE.
- Un cop iniciat el registre d'un producte, aquest ja no es podrà modificar.

Paràmetres d'entrada

- p_idProducte NUMBER: identificador del producte.
- p_registrat NUMBER: estat del registre.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que el producte existeixi a la taula.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: el producte no existeix'
'ERROR: ja existeix un producte amb el mateix: nom, fórmula'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_REGISTRE

Propòsit

- Afegeix un un nou registre a la taula REGISTRE.

Paràmetres d'entrada

- p_producte NUMBER: identificador del producte
- p_pais NUMBER: identificador del país.
- p_distribuidor NUMBER: identificador del distribuïdor.
- p_cost NUMBER: cost del registre.
- p_dataInici DATE: data inici del registre.
- p_dataFi DATE: data fi del registre.
- p_durada NUMBER: duració del tràmits de registre.

Paràmetres de sortida

- RSP VARCHAR2

Verificacions

- Verifica que els camps obligatoris no siguin nuls.
- Verifica que el producte existeix.
- Verifica que el país existeix.
- Verifica que el distribuïdor existeix.

Retorna

- si l'execució ha finalitzat amb èxit:
'OK'
- si l'execució ha fracassat retorna un dels següents missatges:
'ERROR: el camp {camp} no pot ser nul'
'ERROR: {camp} no existeix'
'ERROR: el valor dels camps cost i durada han de ser positius'
'ERROR: {Tipus d'error}' en qualsevol altre cas.

SP_BAIXA_REGISTRE	
Propòsit	<ul style="list-style-type: none"> - Realitza la finalització d'un registre. - S'assigna la data actual com data de fi de registre. - Calculem el temps que ha durat el registre i el seu cost (100€/dia).
Paràmetres d'entrada	<ul style="list-style-type: none"> - p_idRegistre NUMBER: identificador del registre.
Paràmetres de sortida	<ul style="list-style-type: none"> - RSP VARCHAR2
Verificacions	<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que el registre existeixi a la taula.
Retorna	<ul style="list-style-type: none"> - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el registre no existeix' 'ERROR: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_REGISTRE	
Propòsit	<ul style="list-style-type: none"> - Modifica dades d'un registre de la taula REGISTRE. - Un cop s'ha iniciat el registre, només podem modificar la resposta.
Paràmetres d'entrada	<ul style="list-style-type: none"> - p_idRegistre NUMBER: identificador del registre - p_resposta NUMBER: estat del tràmit. - p_cost NUMBER: cost del registre.
Paràmetres de sortida	<ul style="list-style-type: none"> - RSP VARCHAR2
Verificacions	<ul style="list-style-type: none"> - Verifica que els camps obligatoris no siguin nuls. - Verifica que el registre existeixi a la taula.
Retorna	<ul style="list-style-type: none"> - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp {camp} no pot ser nul' 'ERROR: el registre no existeix' 'ERROR: {Tipus d'error}' en qualsevol altre cas.

6.2.2 Mòdul estadístic

La implementació del mòdul estadístic es basa en una sèrie de taules i procediments emmagatzemats a la BD. Cada una de les taules s'utilitza per emmagatzemar la informació requerida per donar resposta a les consultes descrites al punt R6 dels requisits inicials. La informació continguda en aquestes taules es modifica constantment a partir de l'ús dels procediments descrits a l'apartat anterior.

Per tal de que les taules estiguin actualitzades en tot moment, s'han creat altres procediments interns de la BD que s'executen de forma transparent a l'usuari. La seva funció es omplir les taules estadístiques i actualitzar la informació sempre que es produeixin canvis relacionats amb el seu contingut. D'aquesta forma serà possible fer consultes sobre aquestes taules en temps constant 1.

Els procediments que gestionen el mòdul estadístic són els següents:

SP_ALTA_EST1	
Propòsit	<ul style="list-style-type: none"> - Afegeix una entrada a la taula estadística EST1_PROD_COST. - La informació correspon al registre dels productes que costa més registrar (en €). - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada	<ul style="list-style-type: none"> - p_producte NUMBER: identificador del producte. - p_cost NUMBER: cost total del producte.

Verificacions - Verifica que els camps obligatoris no siguin nuls.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST1: el camp {camp} no pot ser nul' 'ERROR EST1: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST1
Propòsit - Actualitza la taula estadística EST1_PROD_COST. - La informació correspon al registre dels productes que costa més registrar (en €). - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada - Cap paràmetre.
Verificacions - Cap verificació.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST1: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_EST2
Propòsit - Afegeix una entrada a la taula estadística EST2_PAISOS_TRAM_LENT. - La informació introduïda correspon als països on els tràmits són més lents. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada - p_pais NUMBER: identificador del país. - p_duradaTramit NUMBER: durada del registre.
Verificacions - Verifica que els camps obligatoris no siguin nuls.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST2: el camp {camp} no pot ser nul' 'ERROR EST2: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST2
Propòsit - Actualitza la taula estadística EST2_PAISOS_TRAM_LENT. - La informació introduïda correspon als països on els tràmits són més lents. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada - Cap paràmetre.
Verificacions - Cap verificació.
Retorna - si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST2: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_EST3
Propòsit - Afegeix una entrada a la taula estadística EST3_TEMPS_MIG. - La informació introduïda correspon al temps mig de registre per país. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada - p_pais NUMBER: identificador del país. - p_mPais NUMBER: temps mig de registre per país.

Verificacions
- Verifica que els camps obligatoris no siguin nuls.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST3: el camp {camp} no pot ser nul' 'ERROR EST3: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST3
Propòsit
- Actualitza la taula estadística EST3_TEMPS_MIG. - La informació introduïda correspon al temps mig de registre per país. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada
- Cap paràmetre.
Verificacions
- Cap verificació.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST3: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_EST4
Propòsit
- Afegeix una entrada a la taula estadística EST4_PERC_DIST_PAISOS. - La informació introduïda correspon al % de països que tenim a la BD que no tenen distribuïdor associat.
Paràmetres d'entrada
- p_paisNoDist NUMBER: % de països que no tenen distribuïdor associat.
Verificacions
- Verifica que els camps obligatoris no siguin nuls.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST4: el camp {camp} no pot ser nul' 'ERROR EST4: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST4
Propòsit
- Actualitza la taula estadística EST4_PERC_DIST_PAISOS. - La informació introduïda correspon al % de països que tenim a la BD que no tenen distribuïdor associat. - Aquest procediment s'executa automàticament quan es dona d'alta, es modifica o es dona de baixa un distribuïdor.
Paràmetres d'entrada
- Cap paràmetre.
Verificacions
- Cap verificació.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST4: {Tipus d'error}' en qualsevol altre cas.

SP_ALTA_EST5
Propòsit
- Afegeix una entrada a la taula estadística EST5_MIG_REG_PAIS. - La informació correspon al número mig de registres per país un any concret. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada
- p_pais NUMBER: identificador del país. - p_anys NUMBER: any de la estadística. - p_mRegistre NUMBER: temps mig de registre per país.

Verificacions
- Verifica que els camps obligatoris no siguin nuls.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST5: el camp {camp} no pot ser nul' 'ERROR EST5: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST5
Propòsit
- Actualitza la taula estadística EST5_MIG_REG_PAIS. - La informació correspon al número mig de registres per país un any concret. - Aquest procediment s'executa automàticament quan es dona de baixa un registre.
Paràmetres d'entrada
- p_pais NUMBER: identificador del país. - p anys NUMBER: any de la estadística.
Verificacions
- Verifica que els camps obligatoris no siguin nuls.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST5: el camp {camp} no pot ser nul' 'ERROR EST5: {Tipus d'error}' en qualsevol altre cas.
SP_ALTA_EST6
Propòsit
- Afegeix una entrada a la taula estadística EST6_FORMAT_NO_REG. - La informació correspon al % de formats de producte que mai s'han registrat.
Paràmetres d'entrada
- p formatNoReg NUMBER: % de formats de producte que mai s'han registrat.
Verificacions
- Verifica que els camps obligatoris no siguin nuls.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST6: el camp {camp} no pot ser nul' 'ERROR EST6: {Tipus d'error}' en qualsevol altre cas.
SP_MODIFICACIO_EST6
Propòsit
- Actualitza la taula estadística EST6_FORMAT_NO_REG. - La informació correspon al % de formats de producte que mai s'han registrat. - Aquest procediment s'executa automàticament quan es dona d'alta, es modifica o es dona de baixa un producte.
Paràmetres d'entrada
- Cap paràmetre.
Verificacions
- Cap verificació.
Retorna
- si l'execució ha finalitzat amb èxit: 'OK'
- si l'execució ha fracassat retorna un dels següents missatges: 'ERROR EST6: {Tipus d'error}' en qualsevol altre cas.

6.2.3 Logs

La implementació del sistema de logs es basa en una taula on s'emmagatzema tota la informació referent a les crides a procediments i funcions del sistema, juntament amb el resultat de l'execució. Per tal de mantenir la taula actualitzada s'ha creat un procediment intern de la BD que s'executa de forma transparent a l'usuari cada vegada que s'utilitzen els procediments i funcions dels apartats anteriors.

Els procediment que actualitza la taula de logs és el següent:

SP_ALTA_LOGS	
Propòsit	
- Afegeix una entrada a la taula LOGS. Aquest procediment s'executa automàticament quan executa qualsevol altre procediment o consulta. La informació afegida correspon al nom del procediment, hora i data, paràmetres d'entrada i paràmetres de sortida (RSP) Paràmetres d'entrada.	
Paràmetres d'entrada	
- p_data DATE: hora i data d'inserció del log. - p_procediment VARCHAR2(200): nom del procediment. - p_paramsEntrada VARCHAR2(2000): paràmetres d'entrada per al procediment cridat. - p_paramsSortida VARCHAR2(2000): paràmetres de sortida per al procediment cridat.	
Verificacions	
- Verifica que els camps obligatoris no siguin nuls.	
Retorna	
- si l'execució ha finalitzat amb èxit: 'OK' - si l'execució ha fracassat retorna un dels següents missatges: 'ERROR: el camp { camp } no pot ser nul' 'ERROR: {Tipus d'error}' en qualsevol altre cas.	

6.3 Proves del sistema

En aquest capítol es mostren proves de funcionament realitzades a la Base de Dades. La realització de les proves es divideix en tres parts: proves de procediments d'alta, baixa i modificació (ABM), proves de les consultes demanades als requisits del projecte i proves de consultes sobre les taules del mòdul estadístic.

Totes les proves que es descriuen a continuació es poden realitzar a la Base de Dades executant els script de proves que s'adjunten amb aquest document. La seva execució genera uns fitxers .log, on es pot observar el resultat d'aquestes proves.

6.3.1 Càrrega inicial de dades

Per a poder realitzar les proves necessàries primer s'ha de fer una càrrega inicial de dades.

S'introdueixen dades a la majoria de les taules del mòdul principal. Com que les insercions es fan utilitzant els procediments d'alta implementats, la resta de taules estadístiques i la taula de logs també s'omplen de dades com a conseqüència de l'execució dels procediments.

Les dades es poden introduir executant l'script `CARREGA_DADES.sql` que s'adjunta amb aquest document. Una vegada realitzada la càrrega de dades ens trobem el següent escenari:

Taula 6: Càrrega de dades

Taula	Quantitat
Marca	10
Envàs	4
Format	10
Presentació	4
Resposta	3
Moneda	16
Continent	6
País	20
Distribuidor	12
Producte	20

6.3.2 Proves de procediments ABM

Després de fer la càrrega inicial, ja estem en condicions de comprovar el funcionament tots els procediments d'alta, ja que el resultat de l'execució ha quedat enregistrat a la taula LOGS.

Com que totes les insercions han estat correctes, es pot observar que al camp *paramsEntrada* sempre apareix el valor 'RSP: OK', que informa de que el resultat de l'execució ha estat correcte.

Per a comprovar la taula de logs es pot utilitzar la sentència següent:

```
SELECT * FROM LOGS ORDER BY 1;
```

Adicionalment es realitzen una sèrie de proves per a mostrar el correcte funcionament dels procediments de baixa i modificació tant en cas d'èxit com en cas d'error.

Es vol deixar constància de que totes les verificacions que es realitzen als procediments han estat provades amb èxit, però per no estendre en excés la memòria i facilitar la lectura, només es mostren proves d'algunes de les situacions d'error més representatives sobre alguns dels elements de la BD:

- Alta correcta d'un nou element.
- Modificació correcta d'un element.
- Baixa correcta d'un element.
- Alta d'un element que ja existeix.
- Modificació que produeix el duplicat d'un altre element de la BD.
- Alta/modificació d'un element que fa referència a un altre element inexistent.
- Alta/baixa/modificació amb valors nuls.
- Alta/modificació d'un element amb valor no admès en algun dels camps.
- Alta/baixa/modificació d'un element que no existeix.

A tall d'exemple es mostren algunes proves realitzades dels procediments ABM de *producte*:

```
--ALTA AMB UN PAIS QUE NO EXISTEIX
SP_ALTA_DISTRIBUIDOR('WorldWide Supply Group', 20, 'NO', RSP);
DBMS_OUTPUT.PUT_LINE(RSP);

--ALTA AMB VALORS INCORRECTES
SP_ALTA_DISTRIBUIDOR('WorldWide Supply Group', 10, '??', RSP);
DBMS_OUTPUT.PUT_LINE(RSP);

--MODIFICACIO QUE DUPLICA UN ALTRE DISTRIBUIDOR
SP_MODIFICACIO_DISTRIBUIDOR(2, 'Catalunya Export', 1, 'NO', RSP);
DBMS_OUTPUT.PUT_LINE(RSP);

--MODIFICACIO AMB VALORS NULS
SP_MODIFICACIO_DISTRIBUIDOR(1, 'Catalunya Export', null, 'SI', RSP);
DBMS_OUTPUT.PUT_LINE(RSP);

--BAIXA D'UN DISTRIBUIDOR QUE NO EXISTEIX
SP_BAIXA_DISTRIBUIDOR(45, RSP );
DBMS_OUTPUT.PUT_LINE(RSP);
```

El resultat complet de les proves es pot observar al fitxer `PROVES_ABM.log` que s'adjunta amb aquest document.

6.3.3 Proves del mòdul estadístic

A continuació es mostra el resultat de les proves realitzades en relació al mòdul estadístic. La realització de les proves consisteix fer consultes sobre la BD en l'estat actual, i posteriorment realitzar alguns canvis per a mostrar que es produeix l'actualització de les estadístiques correctament. Per tant es mostraran els resultats abans i després de fer els canvis.

```
--Realitzem la baixa de cinc REGISTRES
DECLARE
    RSP VARCHAR2(500);
BEGIN
--BAIXES CORRECTES DE REGISTRES
    SP_BAIXA_REGISTRE(1, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(6, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(11, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(12, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(17, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```

R06.1. Els productes que costa més registrar (cal emmagatzemar el cost de cada registre):

```
SELECT * FROM EST1_PROD_COST;
```

R	PRODUCTE	R	COST
1	1		0.09
2	6		0.09
3	11		0.09
4	12		0.09
5	17		0.09

R06.2. Els països a on els tràmits són més lents (considerant el temps entre l'inici dels tràmits de registre fins a la data de resposta final del país):

```
SELECT * FROM EST2_PAISOS_TRAM_LENT;
```

R	PAIS	R	DURADATRAMIT
1	1		0.0009
2	6		0.0009
3	7		0.0009
4	10		0.0009
5	11		0.0009

R06.3. El temps mig de registre per país (mateixes consideracions que el cas anterior):

```
SELECT * FROM EST3_TEMPS_MIG;
```

R	PAIS	R	MPAIS
1	1		0.0009
2	6		0.0009
3	7		0.0009
4	10		0.0009
5	11		0.0009

Podem veure que, com només hem tramitat cinc productes, tots cinc productes tenen el mateix cost i el temps mig de registre per país. Si tramitem més productes (modificant la data del sistema) veurem que tenim unes estadístiques més complexes:

```
(Avancem la data del sistema 15 dies)
--Realitzem la baixa de cinc REGISTRES més
```

```
DECLARE
    RSP VARCHAR2(500);
BEGIN
--BAIXES CORRECTES DE REGISTRES
    SP_BAIXA_REGISTRE(2, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(5, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(8, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(9, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(16, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```

```
(Avancem la data del sistema 15 dies més)
--Realitzem la baixa de cinc nous REGISTRES
```

```
DECLARE
    RSP VARCHAR2(500);
BEGIN
--BAIXES CORRECTES DE REGISTRES
    SP_BAIXA_REGISTRE(3, RSP);
    SP_BAIXA_REGISTRE(3, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(4, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(10, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(19, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    SP_BAIXA_REGISTRE(20, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```

R06.1. Els productes que costa més registrar (cal emmagatzemar-ne el cost total):

```
SELECT * FROM EST1_PROD_COST;
```

R	PRODUCTE	R	COST
1	3	3000.37	
2	20	3000.37	
3	19	3000.37	
4	10	3000.37	
5	4	3000.37	
6	2	1500.32	
7	16	1500.32	
8	9	1500.32	
9	8	1500.32	
10	5	1500.32	

R06.2. Els països a on els tràmits són més lents (considerant el temps entre l'inici dels tràmits de registre fins a la data de resposta final del país):

```
SELECT * FROM EST2_PAISOS_TRAM_LENT;
```

R2	PAIS	R2	DURADATRAMIT
1	1		30.0037
2	11		30.0037
3	6		30.0037
4	5		30.0037
5	2		30.0037
6	3		15.0032
7	9		15.0032
8	8		15.0032
9	4		15.0032
10	7		0.0009

R06.3. El temps mig de registre per país (mateixes consideracions que el cas anterior):

```
SELECT * FROM EST3_TEMPS_MIG;
```

R2	PAIS	R2	MPAIS
1	1		15.0023
2	2		30.0037
3	3		15.0032
4	4		15.0032
5	5		30.0037
6	6		15.0023
7	7		0.0009
8	8		15.0032
9	9		15.0032
10	10		0.0009
11	11		15.0026

R6.5: El número mig de registres per país:

```
SELECT * FROM EST5_MIG_REG_PAIS WHERE anys = 2013 AND pais = 1;
```

R2	PAIS	R2	ANYS	R2	MREGISTRE
1	1		2013		1

El resultat mostra el número mig de registres per país en un any concret. Per a aquest exemple em triat el país 1 (Catalunya) l'any actual (2013). Afegim un registre nou que sigui un producte que es vendrà a Catalunya, finalitzem el registre i en mostrem el nou resultat:

```
DECLARE
    RSP VARCHAR2(500);
BEGIN
    --ALTA CORRECTA D'UN PRODUCTE
    SP_ALTA_PRODUCTE('One new product', 'x7y2 + h3i5', 3, 4, 8, 1, 1, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    --BAIXA CORRECTA D'UN REGISTRE
    SP_BAIXA_REGISTRE(21, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```


R06.5: El número mig de registres per país:

```
SELECT * FROM EST5_MIG_REG_PAIS WHERE anys = 2013 AND pais = 1;
```

PAIS	ANYS	MREGISTRE
1	2013	2.5

Fins aquí hem vist les quatre estadístiques que estan relacionades amb els registres que es duen a terme per a cada país. Ara anem a veure les altres dues estadístiques:

R06.4. Percentatge de països que tenim a la BD que no tenen distribuïdor associat:

```
SELECT * FROM EST4_PERC_DIST_PAISOS;
```

PAISNODIST
0.4

Crearem un nou distribuïdor i en modificarem un altre de manera que ens modifiqui el resultat. Un cop fet això podem tornar a executar els `SELECT * FROM` d'aquesta estadística i fem un altre vegada la consulta:

```
DECLARE
    RSP VARCHAR2(500);
BEGIN
    --MODIFICACIO CORRECTA D'UN DISTRIBUIDOR
    SP_MODIFICACIO_DISTRIBUIDOR(2,'Dubliners - Germany Branch', 2, 'SI', RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    --ALTA CORRECTA D'UN DISTRIBUIDOR
    SP_ALTA_DISTRIBUIDOR('Dubliners - US Branch', 6, 'SI', RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```

R06.4. Percentatge de països que tenim a la BD que no tenen distribuïdor associat:

```
SELECT * FROM EST4_PERC_DIST_PAISOS;
```

PAISNODIST
0.35

R06.6. Percentatge de formats de producte que mai s'han registrat:

```
SELECT * FROM EST6_FORMAT_NO_REG;
```

FORMATNOREG
0.1

Donarem d'alta un nou producte de manera que ens modifiqui el resultat al registrar-ne l'únic format que quedava. Un cop fet això podem tornar a executar els `SELECT * FROM` d'aquesta estadística i fem un altre vegada la consulta:

```
DECLARE
    RSP VARCHAR2(500);
BEGIN
    --ALTA CORRECTA D'UN PRODUCTE
    SP_ALTA_PRODUCTE('Yet another one', 'x3y1 + h2i6', 5, 4, 3, 1, 1, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
    --BAIXA CORRECTA D'UN REGISTRE
    SP_BAIXA_REGISTRE(22, RSP);
    DBMS_OUTPUT.PUT_LINE(RSP);
END;
/
```

R06.6. Percentatge de formats de producte que mai s'han registrat:

```
SELECT * FROM EST6_FORMAT_NO_REG;
```

FORMATNOREG	
1	0

El resultat complet de les proves es pot observar al fitxer `PROVES_ESTADISTIQUES.log` que s'adjunta amb aquest document (les captures de pantalla d'aquest document poden no coincidir amb el fitxer `PROVES_ESTADISTIQUES.log` ja que les proves s'han realitzat diverses vegades).

S'ha de tenir en compte que, un cop fetes les proves dels procediments ABM, s'ha tornat a crear la Base de Dades de nou abans de realitzar les proves del mòdul estadístic.

6.3.4 Comprovació de logs

S'ha adjuntat el contingut de la taula LOGS al final dels fitxers `PROVES_ABM.log` i `PROVES_ESTADISTIQUES.log` resultants de l'execució dels scripts `PROVES_ABM.sql` i `PROVES_ESTADISTIQUES.log` que recull les proves dels procediments ABM i del mòdul estadístic respectivament. D'aquesta forma es poden observar totes les execucions de procediments i funcions produïdes fins a aquest moment, incloent les execucions internes dels procediments de gestió del mòdul estadístic.

També es pot consultar la taula LOGS utilitzant la següent sentència en qualsevol moment:

```
SELECT * FROM LOGS ORDER BY 1;
```

7 Recursos empleats i valoració econòmica

En aquest capítol es detallen els recursos necessaris per dur a terme el projecte a més de donar-ne una valoració econòmica

7.1.1 Recursos humans

Durant la realització d'aquest treball hi participen els següents perfils professionals:

- **Analista:** s'encarrega de totes les tasques relacionades amb les fases d'anàlisi, disseny i documentació del producte.
- **Administrador de Bases de Dades (DBA):** realitza la instal·lació del software SGBD necessari i la creació de la Base de Dades en si.
- **Programador:** s'encarrega de totes les tasques derivades de les fases d'implementació i proves.

7.1.2 Maquinari

S'ha utilitzat en aquest projecte el següent maquinari:

- Portàtil HP Pavilion g6-1240sa:
 - Intel Core i5-2430M Processor (3M Cache, 2.40 GHz) – x64
 - 6 GB de RAM
 - Microsoft Windows 7 Home Premium amb Service Pack 1
- Portàtil IBM Lenovo ThinkPad T420:
 - Intel Core i5-2520M Processor (3M Cache, 2.50 GHz) – x86
 - 4 GB de RAM
 - Microsoft Windows XP Professional Edition amb Service Pack 3

7.1.3 Programari

S'ha utilitzat en aquest projecte el següent programari:

- **Sistema Gestor de Base de Dades (SGBD):** d'acord amb el sol·licitat, la base de dades es desenvoluparà amb Oracle, exactament amb *Oracle Database Express Edition 11g Release 2*.
- **Programari SQL i PL/SQL:** l'aplicació de desenvolupament *Oracle SQL Developer* en la seva versió 3.2.2.
- Disseny del **Diagrama E/R:** Microsoft Visio 2010 i Gliffy Online (editor de diagrames basat en web: <http://www.gliffy.com/>).
- **Planificació** (Diagrama de Gantt): l'aplicació Microsoft Project en la seva versió 2010.

Per a l'elaboració de la **documentació** d'aquest projecte s'han utilitzat les aplicacions ofimàtiques de la família Microsoft Office (Word, Excel, PowerPoint) en la seva versió 2010 i Adobe Reader X.

La planificació i seguiment del projecte es portarà utilitzant Microsoft Project 2010.

7.1.4 Valoració econòmica

La *valoració econòmica* del projecte es realitza amb una previsió de treball de pràcticament dues hores per jornada laboral basada en la **planificació** realitzada.

L'equip del projecte estarà format pels perfils abans esmentats (analista, administrador de Bases de Dades i programador), cadascun d'aquests perfils té associat el següent cost per jornada:

Taula 7: Relació Perfil Professional – Cost – Jornada

Perfil	Cost	Jornades
Analista	120 €	51
Administrador de Bases de Dades (DBA)	100 €	2
Programador	80 €	23

En quant al cost del *Programari* i el *Maquinari*, pressuposem que el client ja té la llicència d'ús per Oracle i que la empresa que fa la implementació (nosaltres) disposa, també, dels recursos necessaris. Per tant, no s'inclou el cost del mateix en la valoració. En tot cas, s'han afegit aquests costos als annexos del document.

Per tant, segons la *Relació Perfil Professional – Cost – Jornada*, i tenint en compte que hi haurà una persona de cada perfil, el cost del desenvolupament, amb l'IVA inclòs, és el següent:

Taula 8: Valoració econòmica

Perfil	Jornades	Cost unitari	Cost total
Analista	51	120 €	6.120 €
Administrador de Bases de Dades	2	100 €	200 €
Programador	23	80 €	1.840 €
		<i>Base</i>	8.160 €
		IVA (21%)	1.713,60 €
		Total	9.873,60 €

8 Conclusions

Després de passar per totes les fases projecte arribem a les següents conclusions:

Podem afirmar que s'han acomplert els objectius especificats a l'inici del projecte ja que, no només m'ha permès posar en pràctica els coneixements adquirits durant assignatures anteriors, sinó que a més considero que he adquirit noves competències quant a la creació i desenvolupament de projectes més complicats dels que havia vist fins al moment. També cal destacar els nous coneixements adquirits amb el sistema de gestió de bases de dades **Oracle**, del que no tenia massa coneixement abans de l'inici del projecte.

S'ha obtingut un producte final de qualitat, ja que després de realitzar les proves hem vist que respon a tots els requeriments especificats pel client.

Per aconseguir això ha sigut molt important l'elaboració d'un pla de treball el més realista possible que m'ha permès avaluar el meu progrés durant les fases del projecte.

Més important encara han sigut les fases d'anàlisi de requisits i la fase de disseny. En un projecte d'aquestes característiques és molt important detallar clarament el que se'ns demana abans de continuar amb la fase següent. La comunicació amb el client (consultor) ha sigut un element molt important ja que han sorgit alguns dubtes sobre els requeriments inicials que gracies a aquesta comunicació s'han aclarit satisfactòriament.

La fase de disseny no ha presentat grans dificultats ja que es disposava d'informació suficient derivada de la fase anterior.

La implementació ha sigut una mica més costosa al començament ja que he hagut d'adaptar-me a una nova eina, però després d'un cert temps tot ha anat correctament. Aquesta capacitat d'adaptació adquirida és potser un dels punts més importants dels objectius assolits al final del projecte.

Com a conclusió final penso que tot i ser un treball didàctic, el projecte actual és un bon punt de partida de cara a futurs projectes professionals fora de l'àmbit acadèmic, ja que ara tinc coneixements de totes les fases necessàries per a realitzar treballs de major envergadura. En general, el resultat ha estat molt satisfactori.

9 Glossari

- **Accés directe:** es tracta d' un concepte íntimament lligat a les bases de dades i a la tecnologia d'emmagatzematge. Quan no existien els discs durs o els formats òptics la informació s'emmagatzemava en cintes o en altres substrats magnètics o físics i, per tant, per a accedir físicament a una dada concreta calia recórrer tota la cinta fins a arribar al punt desitjat. En els formats actuals la informació es localitza mitjançant l'accés directe a la dada perquè no cal recórrer una cinta.
- **ANSI** (American National Standards Institut): és una organització que supervisa el desenvolupament d' estàndards de productes, serveis, processos i sistemes.
- **Atribut:** en el context de les BD, un atribut, és una propietat d'una entitat.
- **Base de Dades Relacional:** és una base de dades que compleix amb el model relacional d'organització de les dades.
- **Clau forana:** en les BD es parla d'una clau forana quan un atribut d'una taula referència a un atribut d'una altra taula.
- **Clau primària:** es defineix en les bases de dades com l'atribut que identifica una entitat i que pren exactament un valor únic per a cada ocurrència.
- **Consistència:** la consistència és una propietat de les Bases de Dades relacionals que ens assegura que no existeixen contradiccions entre les dades i les seves propietats i relacions.
- **Datawarehouse:** és el que en enginyeria de computació es denomina un magatzem de dades. Es tracta d'una col·lecció de dades orientada a un determinat àmbit integrat, no volàtil i variable en el temps, que ajuda a la presa de decisions importants en l'entitat a on s'utilitza.
- **Diagrama E/R:** el diagrama entitat-relació és una eina utilitzada per al modelat de les dades d'un sistema d'informació relacional. Aquest model expressa entitats rellevants d'un sistema, i les seves propietats i interrelacions.
- **Disparadors:** un disparador és un event de programació utilitzat a SQL que fa que una determinada acció s'executi davant d'un succés que intenta modificar l'estat de les dades amb una inserció, una modificació o un esborrat.
- **Disseny conceptual:** és una fase del desenvolupament del programari en què es transformen els conceptes reals en conceptes abstractes mitjançant eines i procediments establerts per la disciplina de la enginyeria informàtica.
- **Disseny lògic:** és justament la fase posterior al disseny conceptual i anterior a la implementació del programari, en què s'estableix la lògica del model i es normalitza la Base de Dades.
- **Domini:** és el conjunt de valors que formen part de la realitat a la que pertany.
- **Eficiència:** en el camp de la informàtica la eficiència expressa la relació entre el resultat obtingut i els recursos utilitzats.
- **Entitat:** una entitat és una representació abstracte d'un objecte del món real que té una sèrie de propietats que es poden representar com a atributs.
- **Entitat-relació:** una entitat-relació o interrelació, és una associació entre entitats.
- **Esborrat en cascada:** es tracta d'un mecanisme proporcionat pels gestors de Bases de Dades relacionals que permet esborrar qualsevol objecte, esborrant automàticament tots els lligams (propietats, relacions, etc.) que estiguin establerts amb aquest objecte.
- **Especificació:** l'especificació és la fase prèvia al disseny conceptual, en la qual es determinen els requeriments del programari a desenvolupar.

- **Estructura estàtica:** una estructura estàtica és aquella que no es pot modificar, només es pot crear i eliminar.
- **Excepció:** les excepcions són les respostes controlades que dona un programa quan s'introdueix un o uns valors d'entrada incorrectes o que no formen part del domini contemplat.
- **Funció:** una funció en el nostre context de les bases de dades és un fragment de programa que davant uns valors determinats d'entrada, els processa i respon amb uns valors determinats de sortida.
- **IBM** (International Business Machines): empresa nord-americana que desenvolupa programes, eines i serveis per a la informàtica des de 1888.
- **Identificador:** normalment parlem d'un identificador quan ens referim a un atribut o conjunt d'atributs que són clau primària d'una taula i per tant identifiquen cada ocurrència.
- **Integritat:** es tracta d'un aspecte fortament lligat amb la consistència de les Bases de dades. És una propietat de les dades de correspondre a representacions plausibles del món real.
- **ISO** (International Organization for Standardization): és l'organisme encarregat de promoure les normes internacionals de fabricació, comerç i comunicació per a totes les branques industrials a excepció de l'elèctrica i l'electrònica.
- **Log:** es tracta d'una taula de la BD que conté atributs que guarden tota la informació de cada transacció que es produeix en la BD. És com un diari històric a on s'enregistren els resultats de les execucions que fa l'usuari.
- **Llenguatge de programació declaratiu:** es tracta d'un tipus de llenguatge de programació en el qual se li diu a la computadora exactament el que es desitja obtenir. L'SQL en els seus fonaments més bàsics és un llenguatge declaratiu que utilitza sentències de consulta (SELECT), actualització (UPDATE) i esborrat (DELETE).
- **Maquinari:** es refereix als elements físics necessaris per a implementar sistemes informàtics (memòria, CPU, impressora, xarxa física, servidors, etc.).
- **Oracle Database Express Edition 11g:** és un SGBD comercial en la seva versió limitada per a ús gratuït, l'hem utilitzat per a desenvolupar el treball que es presenta en aquesta memòria tècnica.
- **Paradigma:** és un model o patró que s'utilitza en una determinada disciplina científica.
- **Paràmetre:** és un valor representatiu d'una variable del programa.
- **PL/SQL:** com que l'SQL era un llenguatge declaratiu estàtic, per a donar dinamisme a la manipulació de dades es va adoptar com a solució el PL/SQL que és una ampliació que permet implementar blocs de sentències i incorpora nous elements com la manipulació de variables, estructures modulars, estructures de control de flux i control d'excepcions.
- **Plataforma:** en el nostre cas parlem de plataforma per a referir-nos a tots els elements software necessaris i sobre els quals ha de funcionar l'SGBD que utilitzem.
- **Procediment emmagatzemat:** els procediments emmagatzemats són programes emmagatzemats físicament en la BD, en el nostre cas implementats en PL/SQL, i que quan es criden s'executen directament en el motor de l'SGBD. Això permet manipular grans quantitats de dades en comunicacions entre servidor i client sense saturar el tràfic de la xarxa de dades sortints i entrants.
- **Programari:** compren tots els components lògics, no físics, que requereix un sistema informàtic per a funcionar correctament.

- **Requeriments:** són les característiques i peculiaritats dels objectes i atributs d'un sistema concret, determinats per la naturalesa pròpia del sistema a desenvolupar. La major part dels requeriments resten ocults a primera vista i és durant la fase d'anàlisi i consultoria quan s'han d'anar descobrint interrogant al client i als usuaris.
- **Requisits:** els requisits són les característiques mínimes del programari i del maquinari que permeten executar la solució implementada.
- **Representacions:** parlem de representacions en el context de les BD per a referir-nos a la conceptualització dels objectes del món real que ens interessin.
- **Registre:** és un bloc que conté un conjunt concret de dades i que es repeteix en una secció continua. Per exemple ens podem referir a un registre com la fila d'una taula o també com un bloc d'informació en un disc dur.
- **Sequel:** és un llenguatge desenvolupat per IBM, precursor de l'SQL.
- **Seqüencial:** que té un ordre establert. En els suports de cinta magnètica, les primeres Bases de Dades emmagatzemaven la informació de forma seqüencial.
- **SGBD:** Sistema de Gestió de Bases de Dades. Són sistemes de programari específics, destinats a fer d'interfície entre la Base de Dades, l'usuari i les aplicacions que la utilitzen.
- **Script:** és un fitxer de text que conté el codi font amb les instruccions necessàries que escriu un programador.
- **SQL Developer:** es tracta d'una aplicació d'entorn gràfic que serveix al programador com a plataforma de desenvolupament del programa amb tot tipus de funcionalitats d'ajuda.
- **Transacció:** una transacció en BD és el fragment de codi que s'executa de manera atòmica, és a dir, una instrucció SQL o un bloc d'instruccions PL/SQL que s'executen d'una sola vegada.
- **Traçabilitat:** es refereix a la facilitat de traducció de les sentències lògiques a les instruccions físiques del programa.

10 Bibliografia

10.1 Materials didàctics d'altres assignatures

- Enginyeria del programari
Jordi Pradel Miquel, Jose Raya Martos
UOC, 1ª Edició (Febrer 2011)
- Ús de bases de dades
Rafael Camps Paré, M. José Casany Guerrero, Jordi Conesa Caralt, Dolors Costal Costa, David Fíguls i Massot, Carme Martín Escofet, Àngels Rius Gavidia, M. Elena Rodríguez González, Toni Urpí Tubella
UOC, 1ª Edició (Febrer 2011)
- Disseny de bases de dades
Xavier Burgués Illa, Blai Cabré i Segarra, Jordi Casas Roma, Dolors Costal Costa, Pere Juanola Juanola, Santiago Ortego Carazo, Joan Anton Pérez Braña, Àngels Rius Gavidia, Ramon Segret i Sala
UOC, 2ª Edició (Setembre 2012)
- Gestió de projectes
José Ramón Rodríguez, Pere Mariné Jové
UOC, 1ª Edició (Setembre 2010)

10.2 Llibres de consulta

- Aprendiendo UML en 24 horas
Joseph Schmuller
Prentice Hall, 1ª Edició (Desembre 2001)
- Oracle Database Express Edition Installation Guide
Diversos autors
Oracle Press, 1ª Edició (Juliol 2001)
- Database Systems: A Practical Approach to Design, Implementation and Management
Thomas M. Connolly, Carolyn E. Begg
Addison Wesley, 4ª Edició (Febrer 2005)

10.3 Enllaços d'internet

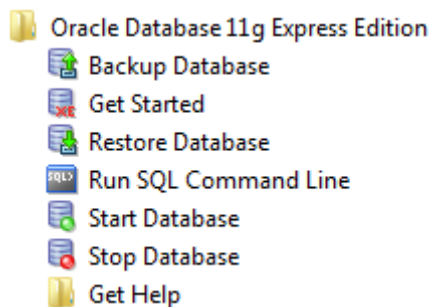
- Oracle Database Express Edition Documentation 11g Release 2 (11.2)
http://docs.oracle.com/cd/E17781_01/index.htm
- Forum: Oracle Database Express Edition (XE)
<https://forums.oracle.com/forums/forum.jspa?forumID=251>
- The Oracle FAQ
<http://www.orafaq.com/>
- PL/SQL Programming Tutorial
<http://www.plsql-tutorial.com/>
- PL/SQL Tutorial
<http://www.tutorialspoint.com/plsql/index.htm>
- Ora Error (errors SQL d'Oracle)
<http://www.ora-error.com/>

11 Annexos

11.1 Manual de funcionament

Tot el codi font del producte s'adjunta amb el document a la carpeta jlopezbon_producte. A partir d'aquest codi font, per implementar la solució, hem de seguir les passes següents:

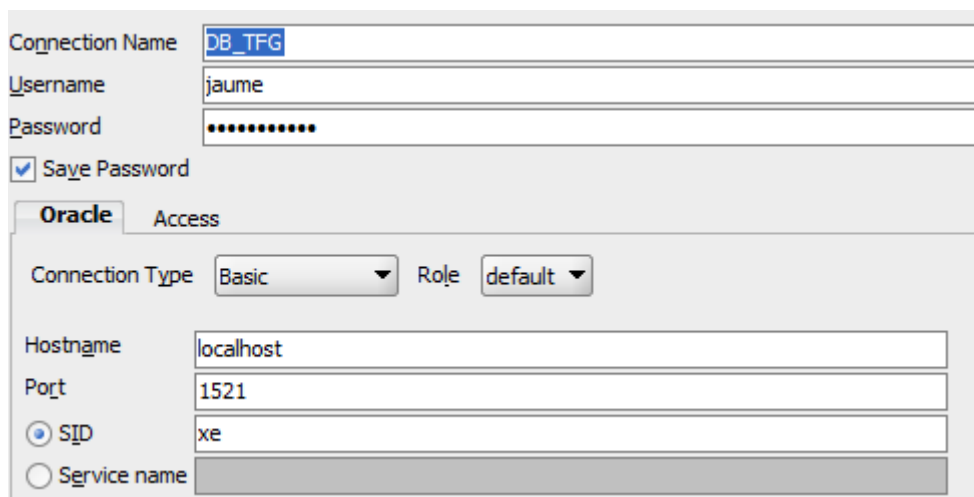
1. El primer pas consisteix en [instal·lar](#) el Sistema de Gestió de Bases de Dades amb el que treballarem, en aquest cas, *Oracle Database 11g Release 2 (11.2.0.1.0)*.
2. Podem iniciar la línia d'ordres escrivint `sqlplus system/password` en una línia de comandes o anar a l'opció *Get Started* al menú de programes, tal com es mostra a continuació:



3. Tot seguit, i per a facilitar la gestió dels scripts, es recomanable instal·lar un editor gràfic per a SQL. En el nostre cas, *Oracle SQL Developer* en la seva versió 3.2.2.
4. Amb l'usuari SYSTEM, executem el codi de creació de l'espai per a taules (*tablespaces*) i de l'usuari, contingut als scripts `0-Tablespaces.sql` i `1-CreacioUsuari.sql` de la carpeta 0-ADMIN.

Si la ruta per als tablespaces és diferent a `C:\oraclexe\app\oracle\oradata\XE`, s'ha de modificar el fitxer `0-Tablespaces.sql`.

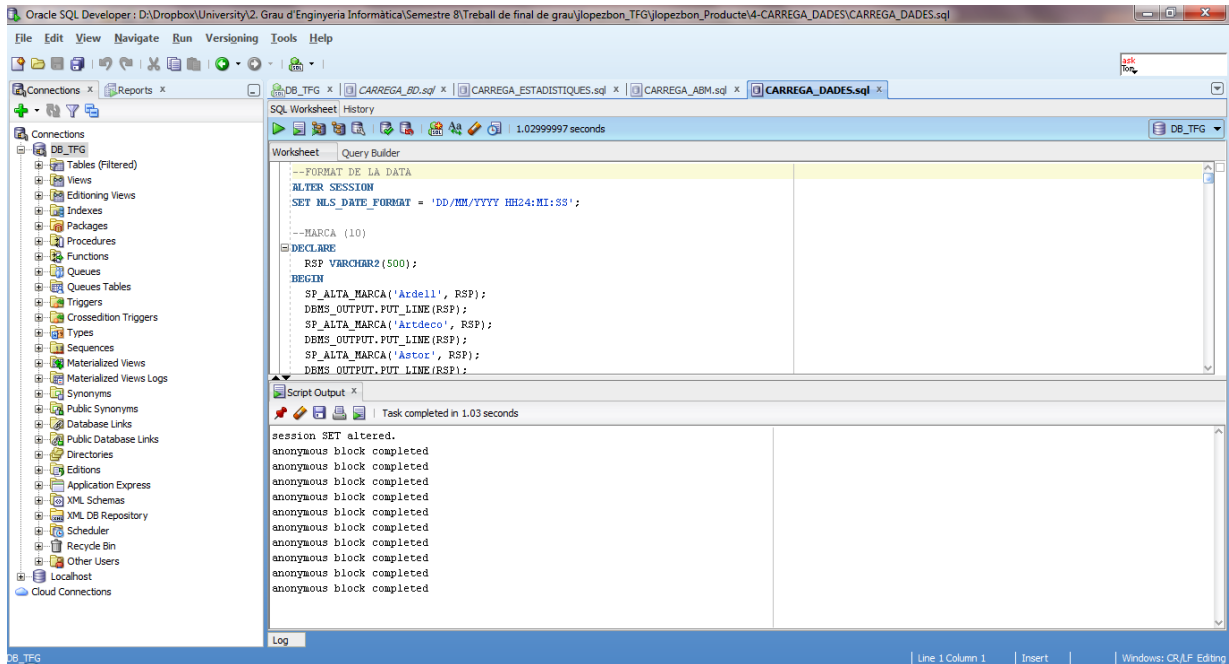
5. Un cop creat l'usuari farem una connexió a la BD via *Oracle SQL Developer*. Cal connectar a la BD amb el compte d'usuari creat (Contrasenya: Gandalf1714):



6. El pas següent és carregar la BD executant el codi contingut a l'script `CARREGA_BD.sql` de la carpeta 1-CARREGA_BD.

Aquest script comporta la creació de totes les taules (`2-CreacioTaules.sql`), els índexs (`3-Index.sql`) i els triggers (`4-Triggers.sql`) i la taula de logs.

7. Després, carreguem el mòdul estadístic executant el codi contingut a l'script `CARREGA_ESTADISTIQUES.sql` de la carpeta 2-ESTADISTIQUES.
8. A continuació s'executen un a un els blocs de creació de procediments emmagatzemats continguts a l'script `CARREGA_ABM.sql` de la carpeta 3-ABM.
9. El següent pas és executar els blocs d'inserció de les dades fictícies per omplir les taules. El codi font es troba a l'script `CARREGA_DADES.sql` ubicat a la carpeta 4-CARREGA_DADES. En el resultat final hauríem de tenir totes les taules carregades d'informació excepte les del mòdul estadístic.



10. Modifiquem l'usuari 'JAUME' per poder realitzar les proves del mòdul estadístic:

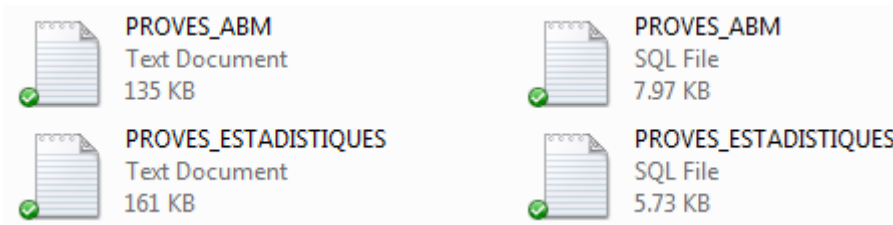
```
-- Connectat com SYSTEM

GRANT ALTER SYSTEM TO JAUME;

COMMIT;
```

11. Per últim farem una sèrie de proves per a verificar el control de les excepcions i l'estat correcte de les dades quan s'insereixen dades al sistema i quan s'executa el procediment d'actualització de les estadístiques.

12. El codi per les proves es troba als scripts de la carpeta 5-PROVES:



11.2 Valoració econòmica dels recursos

Respecte al maquinari utilitzat al projecte, tenim el següent cost associat:

Taula 9: Cost del maquinari

Perfil	Cost
HP Pavilion g6-1240sa	550 €
IBM Lenovo ThinkPad T420	900 €
TOTAL	1.450 €

En quant al programari utilitzat, tenim el següent cost associat (obviant el programari gratuït):

Taula 10: Cost del programari

Perfil	Cost
Oracle Database 11g Standard Edition	4.834 €
Microsoft Office Professional 2010	539 €
Microsoft Project 2010 Professional	1.369 €
Microsoft Visio 2010 Professional	739 €
TOTAL	7.481 €

Per al programari **Oracle Database 11g Standard Edition**, s'ha tingut en compte una llicència per a un (1) *processador* i dos (2) *anys de suport*.

11.3 Instal·lació i configuració del SGBD Oracle

Al descarregar el programari d'Oracle, necessitarem un compte i s'haurà d'acceptar el contracte de llicència.

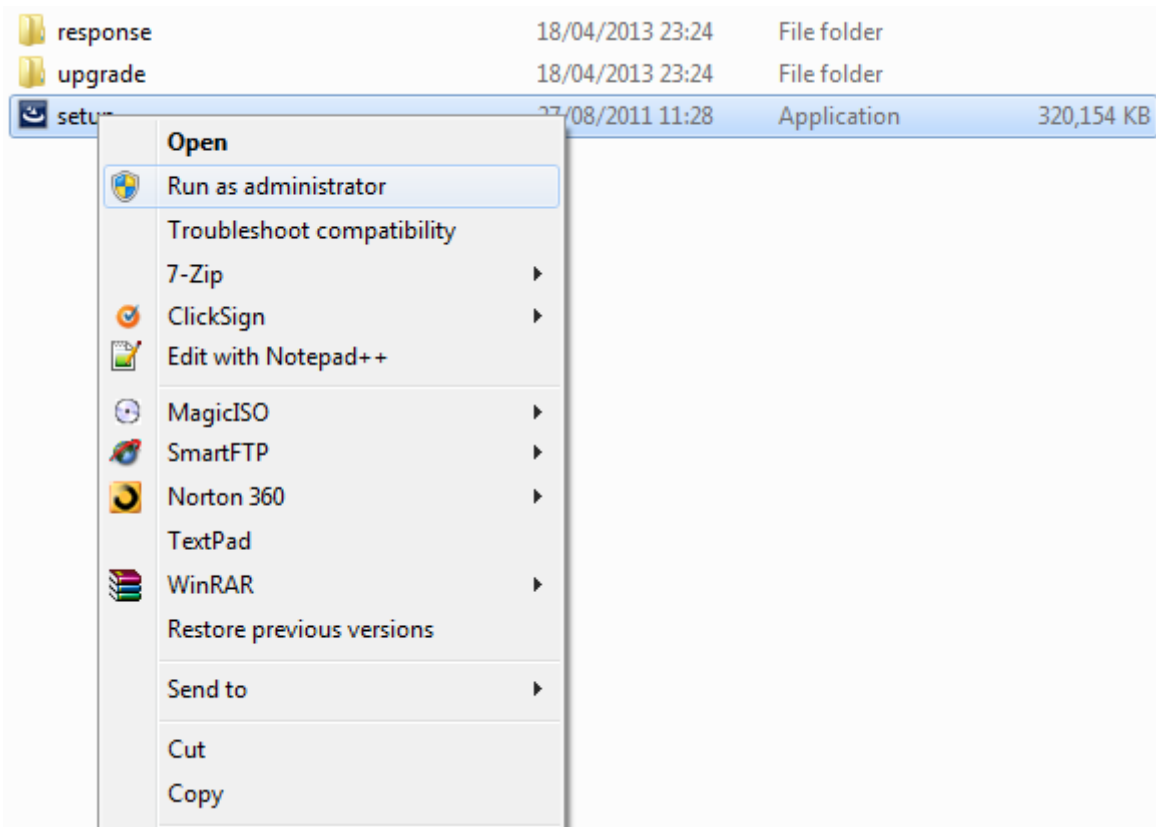
Se'ns demanarà que acceptem l'acord de llicència i no es pot continuar sense fer-ho:



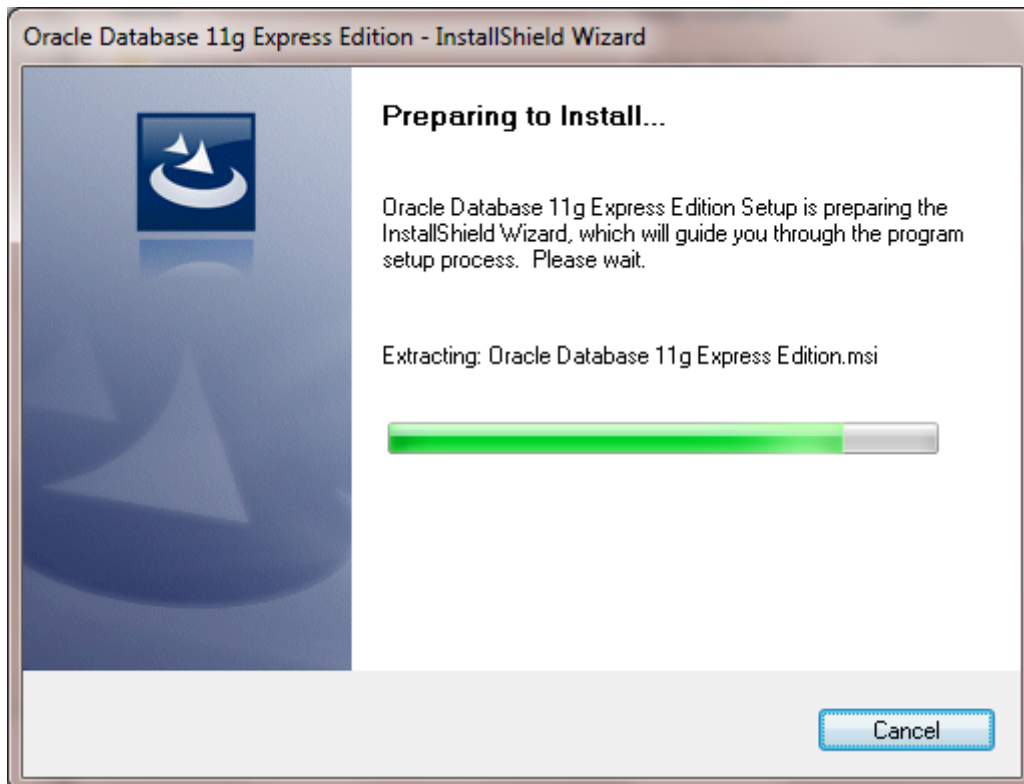
Després d'acceptar l'acord de llicència podem descarregar el programari.

Si ens hi fixem bé, veurem el fitxer que ens ofereixen per baixar és un instal·lador per 32-bit (OracleXE112_Win32.zip), això és degut a que no existeix la versió d'Oracle Express per a sistemes de 64-bit.

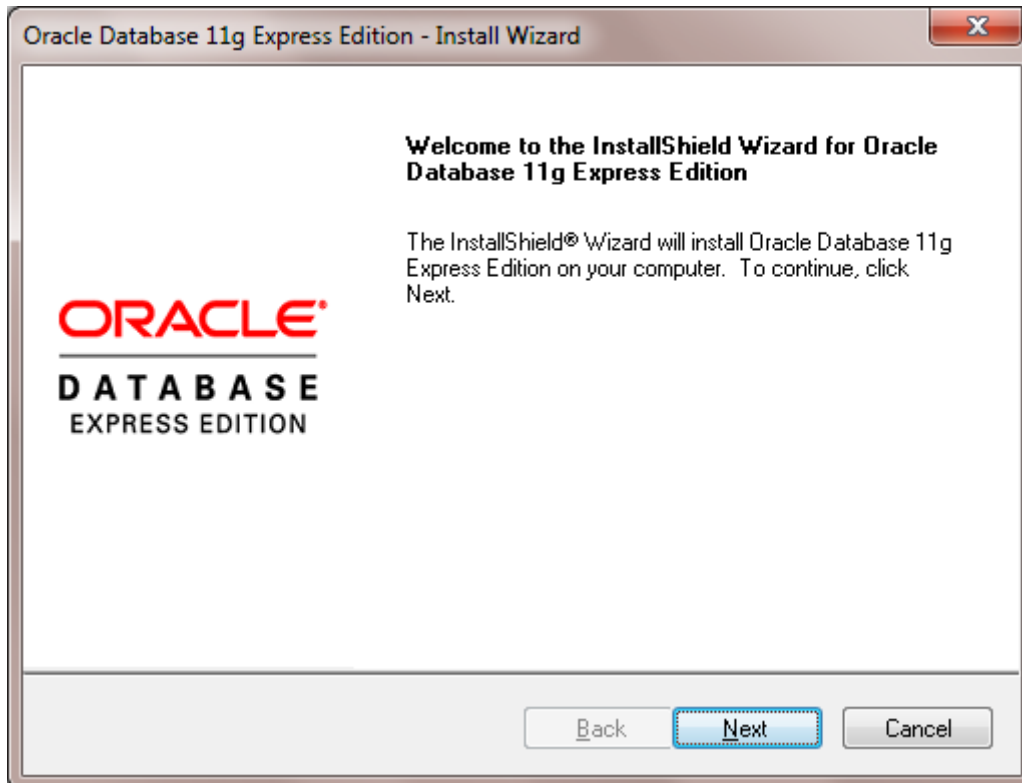
1. Un cop baixat el fitxer el descomprimirem i l'executarem com a administrador:



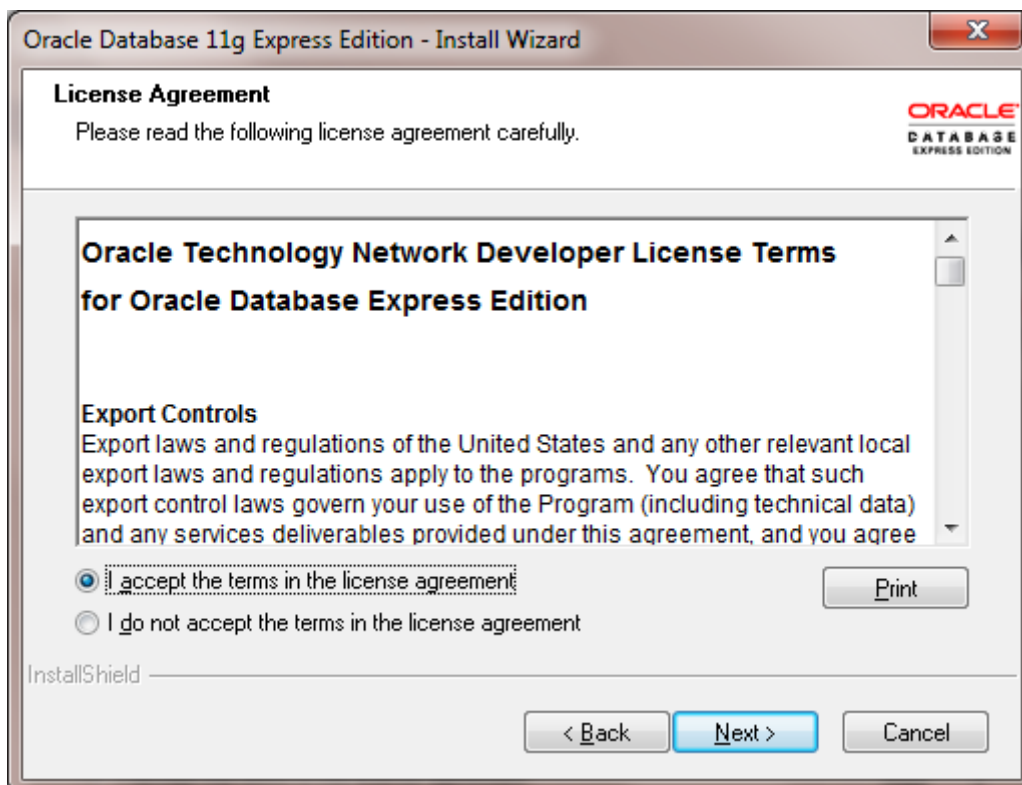
2. Un cop executat el fitxer `setup.msi`, veiem la pantalla del *InstallShield Wizard*:



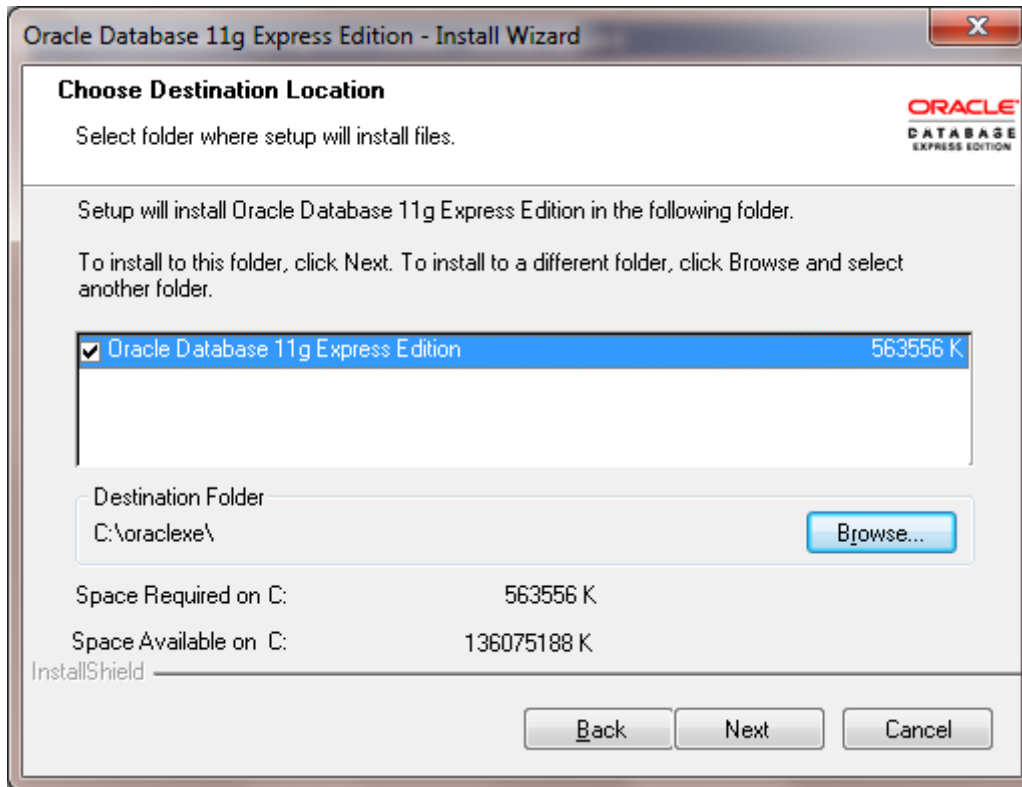
3. I després ja apareix la primera pantalla d'instal·lació. Li donem al botó *Next* per començar la instal·lació.



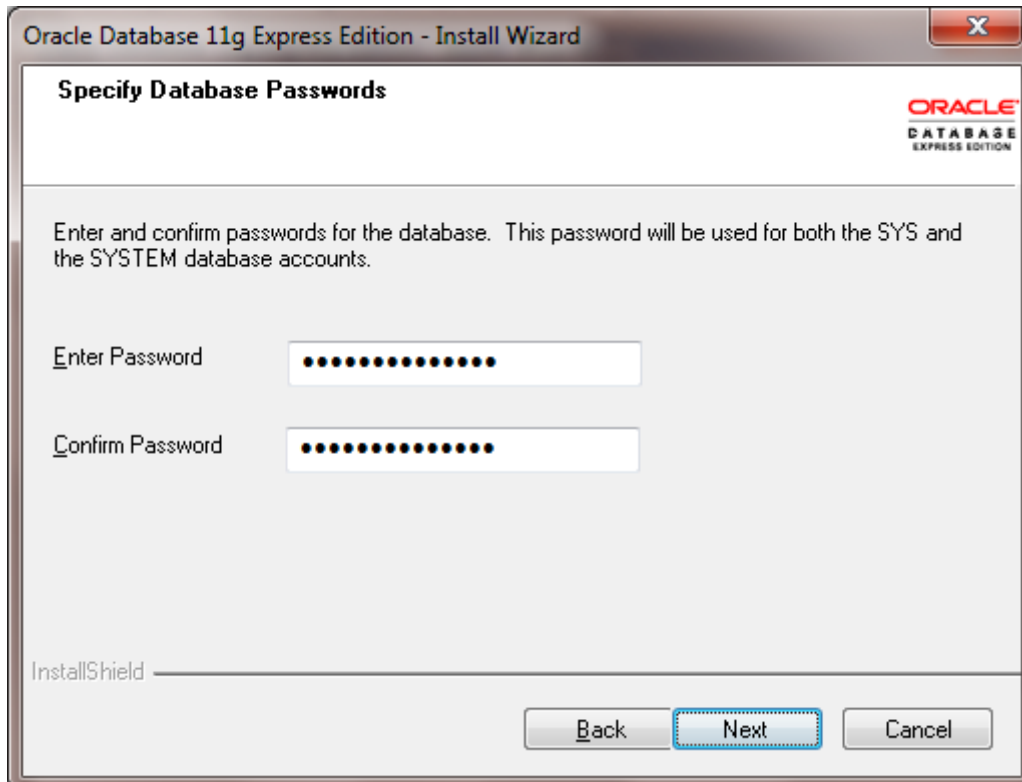
4. A la segona pantalla apareix la llicència de la instal·lació. Marquem *I accept the terms in the license agreement* i després li donem al botó *Next*.



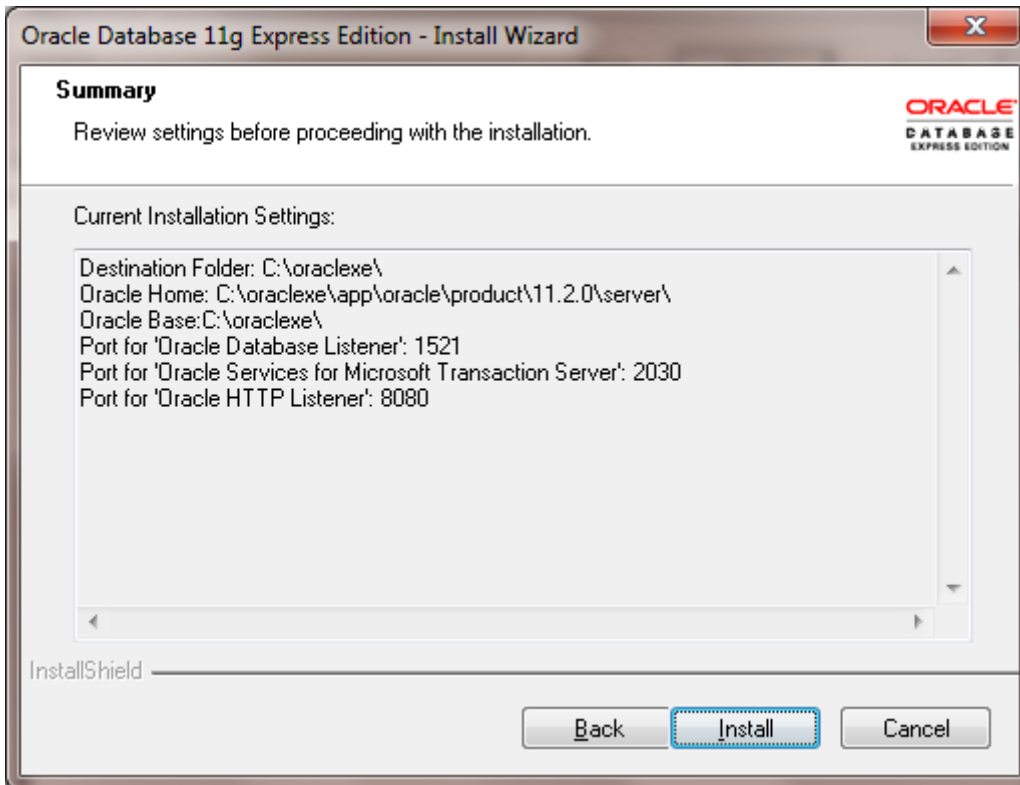
5. A la tercera pantalla acceptarem la localització per defecte i li donem al botó *Next*.



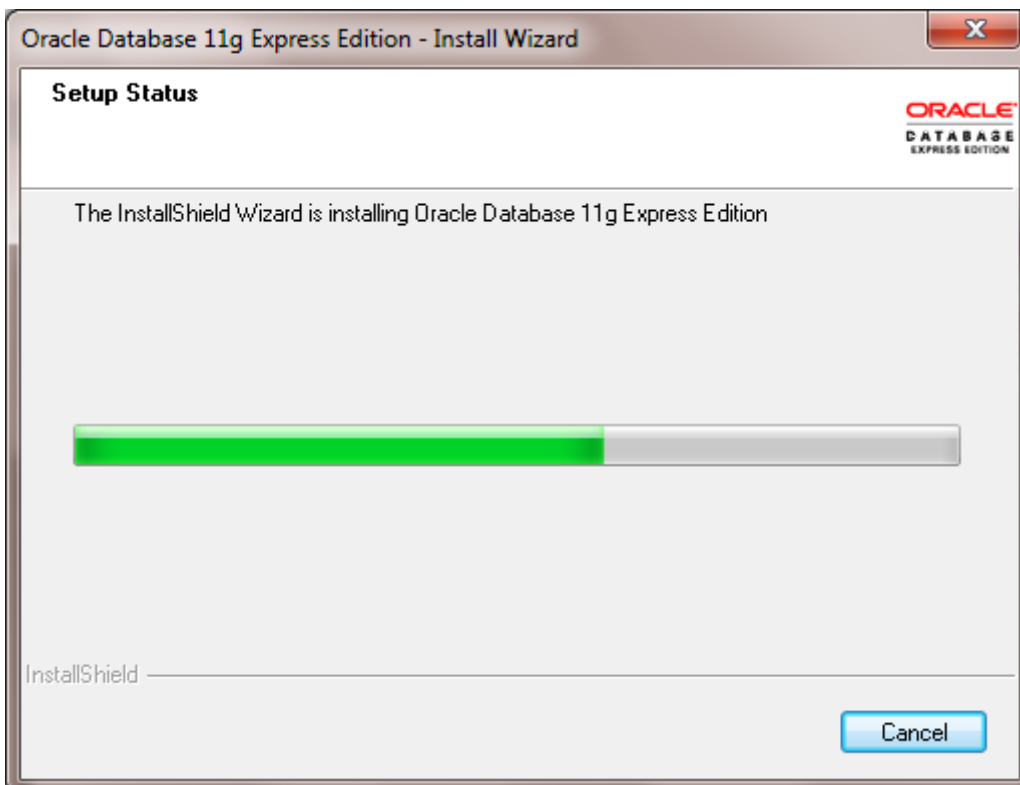
6. A la quarta pantalla introduïrem la contrasenya per SYS i SYSTEM i li donem al botó *Next*.



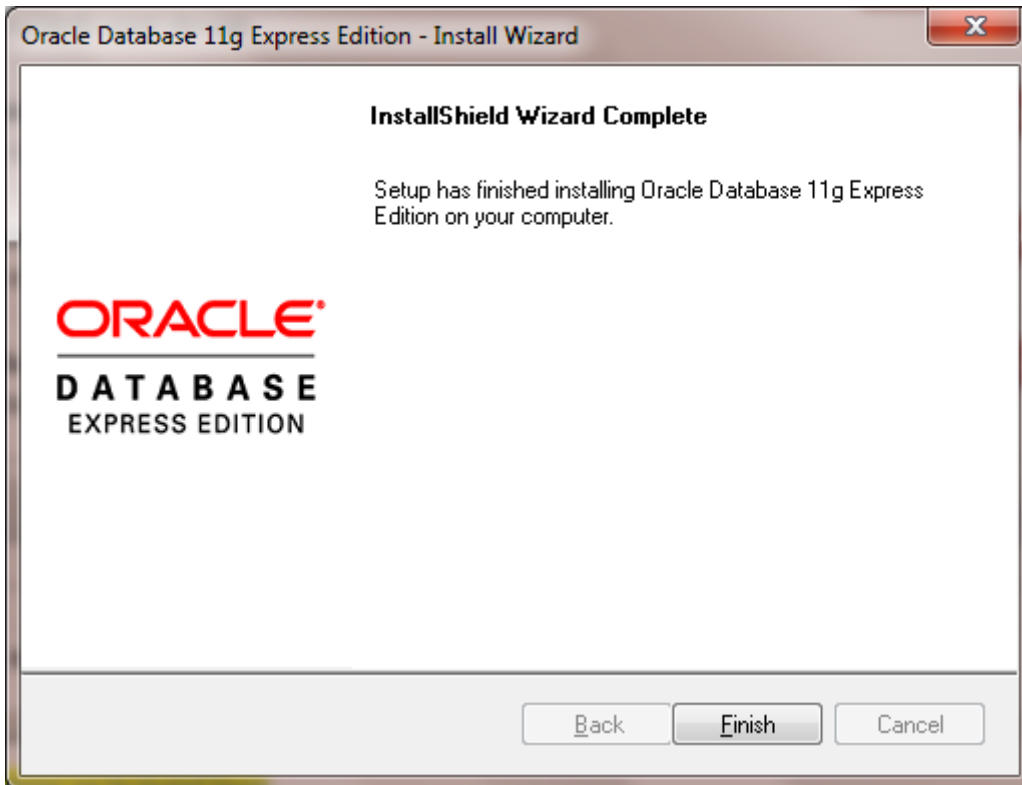
7. La cinquena pantalla ens mostra les opcions triades durant el procés d'instal·lació. Ara ja podem procedir a instal·lar amb el botó *Install*.



8. La sisena pantalla ens mostra el progrés de la instal·lació.



9. A la última pantalla ja ens mostra el final del procés d'instal·lació. Per completar-la li donem al botó *Finish*.



Ara ja s'ha instal·lat Oracle Database 11g Express Edition. Podem iniciar la línia d'ordres escrivint `sqlplus system/password` en una línia de comandes o anar a l'opció *Get Started* al menú de programes.

Si utilitzem, des del menú de programes, *Get Started* veurem la pantalla següent al navegador per defecte que tinguem (en el meu cas Firefox):

