

DISSENY I IMPLEMENTACIÓ DE LA BASE DE DADES D'UN SISTEMA DE CONTROL ENERGÈTIC



Marta Calzado Pamplona
ETIG

Jordi Ferrer Duran

10/06/2012

ÍNDEX

1.	Introducció	5
1.1.	Resum del TFC	5
1.2.	Objectius generals i específics.....	5
1.3.	Documentació a entregar.....	6
1.4.	Enfocament metodològic	6
1.5.	Planificació del projecte	7
1.5.1.	Desglossament de les tasques principals del TFC.....	7
1.5.2.	Planificació PAC's	8
1.5.3.	Planificació Temporal	9
1.6.	Recursos emprats.....	11
1.6.1.	Software	11
1.6.2.	Hardware.....	11
1.6.3.	Humans.....	12
1.7.	Anàlisi i control de riscos	12
1.8.	Valoració econòmica del projecte	12
2.	Anàlisi i Disseny.....	14
2.1.	Anàlisis dels requeriments	14
2.2.	Disseny del Model E/R	18
2.2.1.	Explicació del Model E/R.....	18
2.2.2.	Consideracions no reflectides al diagrama	20
2.2.3.	Atributs de les entitats	21
2.3.	Disseny del Model Lògic.....	22
2.3.1.	Model Relacional	22
2.3.2.	Restriccions d'integritat referencial	24
2.3.3.	Diagrama Lògic Relacional.....	25
2.4.	Disseny Físic	26
2.4.1.	Creació de la Base de Dades	26
2.4.2.	Tablespaces	26
2.4.3.	Usuaris	26
2.4.4.	Taules.....	26
2.4.5.	Seqüències.....	28
2.4.6.	Triggers	28

3.	Implementació	29
3.1.	Procediments d'alta, baixa i modificació	29
3.2.	Consultes	49
3.3.	Estadístiques	52
4.	Seguiment i millores del TFC	57
4.1.	Possibles millores	59
5.	Joc de proves	59
6.	Conclusió	61
7.	Glosari	62
8.	Bibliografia	63
9.	Annexos	63

*Als meu pares, Luis i Conchita, per la paciència
i dedicació que han tingut sempre amb nosaltres.*

*Sense el seu sacrifici i esforç avui no estaria
on estic ni seria qui sóc.*

Gràcies.

1. Introducció

1.1. Resum del TFC

Donada la necessitat a nivell europeu d'aconseguir un control energètic, la Comunitat Europea vol implementar una aplicació que permeti la generació de dades estadístiques de l'ús de l'energia.

El projecte consisteix en desenvolupar la primera fase d'aquesta aplicació. Per dur a terme aquesta fase, la Comunitat Europea ha obert un concurs públic que consisteix en realitzar el disseny d'una Base de Dades que permeti emmagatzemar tota la informació necessària per la generació de dades estadístiques sobre l'ús de l'energia. En una segona fase es desenvoluparà l'aplicació de gestió del control energètic.

A nivell general, la funció de la Base de Dades ha de ser la de guardar tota la informació necessària per entendre i fer actuacions per millorar el consum energètic, així com per a guardar l'històric del mateix. De la mateixa manera, ha de permetre guardar la informació dels clients, comptadors, centraletes de distribució, centrals de producció, etc., permetint realitzar amb aquesta informació diferents consultes habituals com el consum mig de tots els clients, la línia que ha estat més carregada a nivell d'energia consumida, etc.

A més, la Base de Dades ha de permetre precalcular i emmagatzemar diferents informacions estadístiques a partir de procediments, ja implementats en el projecte, que augmentin l'eficàcia i rapidesa del sistema.

1.2. Objectius generals i específics

L'Objectiu principal del TFC és aconseguir desenvolupar un projecte que permeti posar en pràctica i ampliar tots els coneixements adquirits al llarg dels estudis d'Enginyeria Tècnica en Informàtica de Gestió. En aquest projecte es fa èmfasi principalment en les assignatures de Base de Dades I, Base de Dades II, Sistema de Gestió de Base de Dades i Estructura de la Informació.

Un altre objectiu genèric important és demostrar la capacitat per a realitzar un projecte en la seva totalitat, és a dir, gestionar i desenvolupar un treball en totes les seves etapes que van des de l'anàlisi previ dels objectius, fins l'entrega final del producte. I demostrar la capacitat de realitzar totes les parts del projecte en el rol corresponent: Cap de Projecte, Analista, Programador i Testejador.

Els objectius específics del TFC són els següents:

- Adquirir coneixements i destresa amb el sistema de gestió de Base de Dades Relacional que per aquest projecte farem servir el que s'anomena Oracle.
- Aprendre i afiançar els coneixements amb el llenguatge PL/SQL que serà l'eix principal del TFC.
- Aprofundir en el coneixement del llenguatge SQL
- Realitzar i entregar tres proves d'avaluació continuada que son: PAC1, PAC2 i PAC3.
- Realitzar i entregar el TFC finalitzat amb tots els documents pertinents com son: la Memòria, Presentació i el producte final que és la Base de Dades.

1.3. Documentació a entregar

El TFC de Base de Dades relacionals està format per la següent documentació:

- **Pla de treball:** planificació de les tasques que es duran a terme al llarg del desenvolupament del projecte.
- **Producte:** Base de Dades demanada pel client que comprèn els scripts de creació de la BD en llenguatge PL/SQL, procediments emmagatzemats, disparadors, etc.
- **Jocs de prova:** conjunt de proves on es comprovarà el correcte funcionament de la Base de Dades.
- **Memòria:** document de síntesi final del projecte.
- **Presentació:** resum del projecte realitzat on es demostra els resultats obtinguts.
- **Debat Virtual:** es resoldran les diferents qüestions formulades pel tribunal.

1.4. Enfocament metodològic

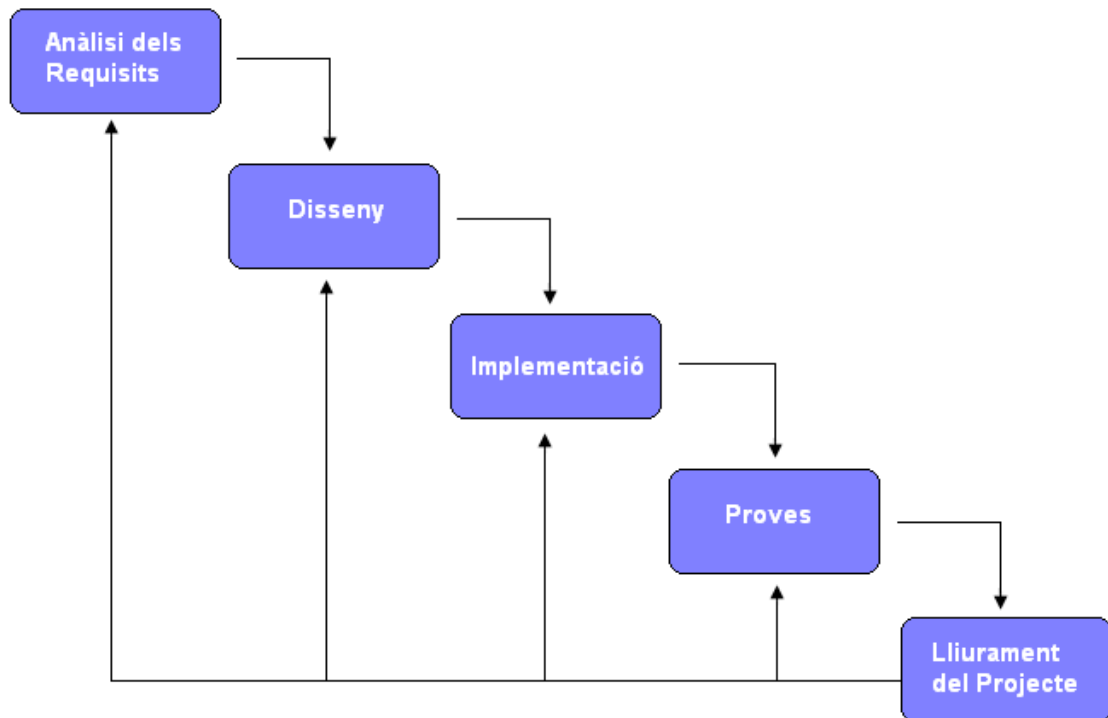
És important fer una bona distribució de les tasques a realitzar al llarg de la duració del projecte des del seu inici fins a la data clau 10/06/2012, data on s'ha de fer el lliurament, per tal d'entregar un producte amb garanties.

Així doncs, donades les característiques del projecte de mitjana complexitat i dels mitjans humans i tècnics disponibles, s'ha optat per fer servir un model de cicle de vida de software en Cascada (Vegeu Imatge 1).

S'ha de tenir present que en aquest projecte una sola persona haurà d'agafar els diferents rols participants així com Cap de projecte, Analista, Programador i finalment Testejador del producte final.

Per dur a terme la realització del treball, primerament es farà un anàlisi de tots els requisits per poder fer un bon disseny del producte. Un cop es disposa del disseny passem a desenvolupar el projecte, i finalment, es faràn diverses proves on s'intentarà verificar que tots els requisits demanats pel client es fan presents.

CICLE DE VIDA EN CASCADA



Imatge1. Cicle de vida en Cascada

1.5. Planificació del projecte

1.5.1. Desglossament de les tasques principals del TFC

Les tasques principals que componen el projecte són les següents:

- Planificació del TFC
 - Lectura i comprensió de l'enunciat del TFC
 - Cercar el software necessari per l'elaboració del projecte
 - Descomposar el projecte en les diferents tasques
 - Planificació temporal de les tasques del TFC
 - Elaboració del pla de treball

- Anàlisis i disseny de la base de dades
 - Anàlisi dels requeriments sol·licitats pel client
 - Instal·lació i configuració del SGBD Oracle
 - Disseny del model E/R de la Base de Dades
 - Disseny del model lògic de la Base de Dades
 - Disseny del model físic de la Base de Dades

- Implementació i desenvolupament de la Base de Dades
 - Creació dels scripts de la Base de dades
 - Creació dels disparadors de la Base de Dades
 - Creació dels procediments emmagatzemats de la Base de Dades
 - Creació del mòdul estadístic
- Proves
 - Proves unitàries dels scripts
 - Proves unitàries dels disparadors
 - Proves unitàries dels procediments
 - Creació d'un joc de proves per verificar que funciona correctament i s'adapta a les necessitats del client
- Lliurament del projecte final
 - Elaboració de la memòria que algunes parts s'hauran anat complimentant al llarg de les entregues de les diferents PAC's.
 - Creació de la presentació del projecte final
 - Revisió i entrega final del projecte incloent memòria, presentació i la Base de Dades
- Debat virtual
 - Respondre les qüestions plantejades pel tribunal sobre el projecte presentat.

1.5.2. Planificació PAC's

La següent taula mostra les dates claus del projecte:

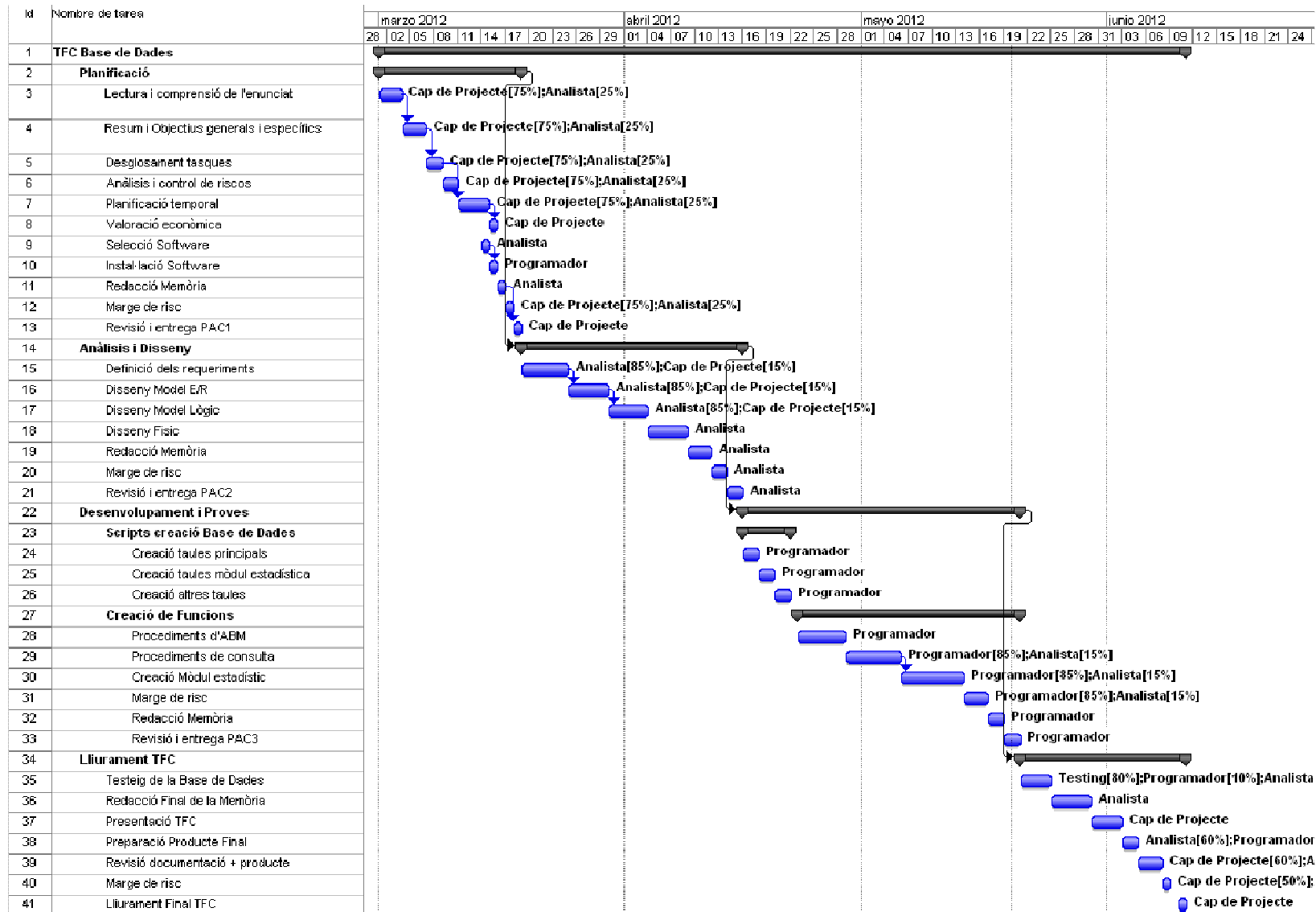
DATA	LLIURAMENT	DESCRIPCIÓ
18/03/2012	PAC1	Pla de treball
15/04/2012	PAC2	Definició dels requeriments Anàlisi i Disseny de la BD Desenvolupament d'una part de la memòria
20/05/2012	PAC3	Scripts de creació de BD Creació procediments Creació disparadors Creació Mòdul Estadístic Creació joc de proves Desenvolupament d'una part de la memòria
10/06/2012	Lliurament final TFC	Base de Dades del projecte Memòria Presentació
A partir del lliurament final del TFC	Debat Virtual	Resoldre les qüestions que exposi el tribunal

1.5.3. Planificació Temporal

S'han de tenir en compte totes les tasques del projecte. La següent taula mostra una planificació temporal des de l'inici del treball fins al seu lliurament i els recursos que es faran servir en cada una d'elles:

	Nombre de tarea	Duración	Comienzo	Fin	Nombres de los recursos
1	TFC Base de Dades	102 dies	jue 01/03/12	dom 10/06/12	
2	Planificació	18 dies	jue 01/03/12	dom 18/03/12	
3	Lectura i comprensió de l'enunciat	3 dies	jue 01/03/12	sáb 03/03/12	Cap de Projecte[75%];Analista[25%]
4	Resum i Objectius generals i específics	3 dies	dom 04/03/12	mar 06/03/12	Cap de Projecte[75%];Analista[25%]
5	Desglossament tasques	2 dies	mié 07/03/12	jue 08/03/12	Cap de Projecte[75%];Analista[25%]
6	Anàlisi i control de riscos	2 dies	vie 09/03/12	sáb 10/03/12	Cap de Projecte[75%];Analista[25%]
7	Planificació temporal	2 dies	dom 11/03/12	lun 12/03/12	Cap de Projecte[75%];Analista[25%]
8	Valoració econòmica	1 dia	mar 13/03/12	mar 13/03/12	Cap de Projecte
9	Selecció Software	1 dia	mié 14/03/12	mié 14/03/12	Analista
10	Instal·lació Software	1 dia	jue 15/03/12	jue 15/03/12	Programador
11	Redacció Memòria	1 dia	vie 16/03/12	vie 16/03/12	Analista
12	Marge de risc	1 dia	sáb 17/03/12	sáb 17/03/12	Cap de Projecte[75%];Analista[25%]
13	Revisió i entrega PAC1	1 dia	dom 18/03/12	dom 18/03/12	Cap de Projecte
14	Anàlisi i Disseny	28 dies	lun 19/03/12	dom 15/04/12	
15	Definició dels requeriments	7 dies	lun 19/03/12	dom 25/03/12	Analista[85%];Cap de Projecte[15%]
16	Disseny Model ER	7 dies	lun 26/03/12	dom 01/04/12	Analista[85%];Cap de Projecte[15%]
17	Disseny Model Lògic	7 dies	lun 02/04/12	dom 08/04/12	Analista[85%];Cap de Projecte[15%]
18	Redacció Memòria	3 dies	lun 09/04/12	mié 11/04/12	Analista
19	Marge de risc	2 dies	jue 12/04/12	vie 13/04/12	Analista
20	Revisió i entrega PAC2	2 dies	sáb 14/04/12	dom 15/04/12	Analista
21	Desenvolupament i Proves	35 dies	lun 16/04/12	dom 20/05/12	
22	Scripts creació Base de Dades	6 dies	lun 16/04/12	sáb 21/04/12	
23	Creació taules principals	2 dies	lun 16/04/12	mar 17/04/12	Programador
24	Creació taules mòdul estadística	2 dies	mié 18/04/12	jue 19/04/12	Programador
25	Creació altres taules	2 dies	vie 20/04/12	sáb 21/04/12	Programador
26	Creació de Funcions	29 dies	dom 22/04/12	dom 20/05/12	
27	Procediments d'ABM	6 dies	dom 22/04/12	vie 27/04/12	Programador
28	Procediments de consulta	6 dies	sáb 28/04/12	jue 03/05/12	Programador[85%];Analista[15%]
29	Creació Mòdul estadístic	6 dies	vie 04/05/12	mié 09/05/12	Programador[85%];Analista[15%]
30	Testeig de la Base de Dades	4 dies	jue 10/05/12	dom 13/05/12	Testing[80%];Programador[10%];Analista[10%]
31	Marge de risc	3 dies	lun 14/05/12	mié 16/05/12	Programador[85%];Analista[15%]
32	Redacció Memòria	2 dies	jue 17/05/12	vie 18/05/12	Programador
33	Revisió i entrega PAC3	2 dies	sáb 19/05/12	dom 20/05/12	Programador
34	Lliurament TFC	21 dies	lun 21/05/12	dom 10/06/12	
35	Redacció Final de la Memòria	6 dies	lun 21/05/12	sáb 26/05/12	Analista
36	Presentació TFC	5 dies	dom 27/05/12	jue 31/05/12	Cap de Projecte
37	Preparació Producte Final	4 dies	vie 01/06/12	lun 04/06/12	Analista[60%];Programador[40%]
38	Revisió documentació + producte	3 dies	mar 05/06/12	jue 07/06/12	Cap de Projecte[60%];Analista[40%]
39	Marge de risc	1 dia	vie 08/06/12	vie 08/06/12	Cap de Projecte[50%];Analista[30%];Programador[20%]
40	Lliurament Final TFC	1 dia	dom 10/06/12	dom 10/06/12	Cap de Projecte

El següent diagrama de Gantt mostra la planificació temporal del projecte:



El diagrama de Gantt anteriorment exposat és seqüencial, ja que una mateixa persona exerceix els diferents rols en un mateix projecte. Les tasques no es poden solapar, és a dir, no es poden fer dues tasques en un mateix moment.

1.6. Recursos emprats

Per dur a terme la realització del projecte es faran servir diferents recursos a nivell de software, hardware i recursos humans. Queden especificats els diferents tipus de recursos en els següents apartats.

1.6.1. Software

A nivell de software el recursos emprats són els següents:

- Oracle Express v10.2.0.1 → com a Sistema de Gestió de Base de Dades (SGBD)
- SQL Developer → per a desenvolupar el projecte en el llenguatge SQL i PL/SQL així com per a realitzar les proves, depuració d'errors i optimització.
- MagicDraw → per a la realització dels diagrames Entitat-Relació (E/R)
- Microsoft Word 2007 → processador de textos per a realitzar tota la documentació com és la memòria, pla de treball, etc.
- Microsoft Power Point → per a realitzar la presentació virtual del projecte
- PDF Creator → per a fer la conversió dels documents a format .pdf
- MSProject 2007 → per a realitzar el diagrama de la planificació temporal (Diagrama de Gantt)

Aquesta seria la previsió de software necessària. Malgrat tot, deixariem les portes obertes a d'altres alternatives equivalents a les triades inicialment. Com per exemple: en el disseny del diagrama E/R també es podria fer servir el Microsoft Visio 2003 o el TOAD per a desenvolupar el projecte.

En qualsevol cas es farà servir aquell software que sigui millor per presentar el projecte en la seva màxima qualitat.

1.6.2. Hardware

A nivell de Hardware el recursos emprats són els següents:

- Ordinador portàtil ACER amb processador Intel Core Duo, 1.73Ghz, 120Gb de disc dur i 1Gb de memòria RAM i Sistema Operatiu Microsoft Windows XP Professional 2002 Service Pack3.

Aquesta màquina exercirà tant de servidor com de client de la Base de Dades.

1.6.3. Humans

Els diferents rols que intervenen en el procés de creació del projecte els farà una mateixa persona (en aquests cas l'alumne). Així, segons la tasca requerida en cada moment adoptarà el paper segons convingui: Cap de Projecte, Analista, Programador i Testejador de l'aplicatiu.

El consultor del projecte agafarà el rol de Client.

1.7. Anàlisi i control de riscos

Durant el procés de desenvolupament d'un projecte ens podem trobar en diferents situacions inesperades que poden interrompre el ritme de treball, fins i tot, ocasionar danys irreparables com la pèrdua del projecte.

Es poden contemplar diferents tipus de riscos al llarg de tot el procés com poden ser:

- Falta de coneixements dels programes que es faran servir i que poden ocasionar retards en la planificació inicial del projecte
- Problemes humans com enfermetats, accidents, etc.
- Incidents a nivell tècnic com ser atacat per un virus informàtic que pot ocasionar pèrdues d'informació, fallada del disc dur, etc.

Per tal d'evitar riscos d'aquests tipus es prendran les següents mesures:

1. En la planificació temporal del projecte, s'han deixat uns dies de marge de seguretat per cada PAC (Marge de risc). D'aquesta manera, en el cas que sorgís qualsevol de les circumstàncies esmentades anteriorment o haguéssim fet una mala planificació temporal, disposariem d'aquests dies per esmentar l'error i arribar a les dates claus.
2. Pel que respecta als imprevistos a nivell tècnic, es farà una còpia de seguretat diària en un disc dur extern així com en un dispositiu USB.

1.8. Valoració econòmica del projecte

En aquest apartat es fa una valoració econòmica del projecte. Per dur a terme la valoració s'han fet quatre taules:

- Taula amb el preu/hora de cada recurs.
- Taula amb la descripció de les tasques, duració, recurs o recursos assignats i el preu corresponent per dia.
- Taula resum amb les tasques principals i el preu total del projecte.
- Taula amb el preu total incloent un 100% de Marge de risc.

La valoració s'ha fet amb una jornada laboral de dilluns a diumenge treballant dues hores al dia.

- Taula amb el preu/hora (net) per recurs:

RECURS	PREU
Cap de Projecte	28€/h
Analista	25€/h
Programador	22€/h
Testing	17€/h

- Taula amb la descripció de les tasques, duració, recurs assignat i preu:

TASCA	DURACIÓ	RECURS ASSIGNAT	PREU
Planificació	18 dies (36h)		947,50€
Lectura i comprensió de l'enunciat	3 dies	Cap de Projecte[75%];Analista[25%]	163,50€
Resum i Objectius generals i específics	3 dies	Cap de Projecte[75%];Analista[25%]	163,50€
Desglosament de tasques	2 dies	Cap de Projecte[75%];Analista[25%]	121,50€
Anàlisi i control de riscos	2 dies	Cap de Projecte[75%];Analista[25%]	121,50€
Planificació temporal	2 dies	Cap de Projecte[75%];Analista[25%]	121,50€
Valoració econòmica	1 dia	Cap de Projecte	56,00€
Selecció Software	1 dia	Analista	50,00€
Instal·lació Software	1 dia	Programador	44,00€
Redacció Memòria	1 dia	Analista	50,00€
Marge de risc	1 dia	Cap de Projecte[75%];Analista[25%]	-
Revisió i entrega PAC1	1 dia	Cap de Projecte	56,00€
Anàlisi i Disseny	28 dies (56h)		1314,40€
Definició dels requeriments	6 dies	Analista[85%];Cap de Projecte[15%]	305,40€
Disseny Model E/R	5 dies	Analista[85%];Cap de Projecte[15%]	254,50€
Disseny Model Lògic	5 dies	Analista[85%];Cap de Projecte[15%]	254,50€
Disseny Físic	5 dies	Analista	250,00€
Redacció Memòria	3 dies	Analista	150,00€
Marge de risc	2 dies	Analista	-
Revisió i entrega PAC2	2 dies	Analista	100,00€
Desenvolupament i Proves	35 dies (70h)		1389,20€
Scripts creació Base de Dades	6 dies (12h)		264,00€
Creació taules principals	2 dies	Programador	88,00€
Creació taules mòdul estadística	2 dies	Programador	88,00€
Creació altres taules	2 dies	Programador	88,00€
Creació de Funcions	29 dies (58h)		1125,20€
Procediments d'ABM	6 dies	Programador	264,00€
Procediments de consulta	6 dies	Programador[85%];Analista[15%]	269,40€
Creació Mòdul estadístic	6 dies	Programador[85%];Analista[15%]	269,40€
Testeig de la Base de Dades	4 dies	Testing[80%];Programador[10%];Analista[10%]	146,40€
Marge de risc	3 dies	Programador[85%];Analista[15%]	-
Redacció Memòria	2 dies	Programador	88,00€
Revisió i entrega PAC3	2 dies	Programador	88,00€
Lliurament TFC	21 dies (42h)		987,40€
Redacció Final de la Memòria	6 dies	Analista	300,00€
Presentació TFC	5 dies	Cap de Projecte	280,00€
Preparació Producte Final	4 dies	Analista[60%];Programador[40%]	190,40€
Revisió documentació + producte	3 dies	Cap de Projecte[60%];Analista[40%]	161,00€

Marge de risc	2 dies	Cap de Projecte[50%];Analista[30%]; Programador[20%]	-
Lliurament Final TFC	1 dies	Cap de Projecte	56,00€

➤ Taula resum de les tasques principals, dies treballats i preus totals:

TASCA	DIES (Hora)	PREU
Planificació	18 dies (36h)	947,50€
Anàlisi i Disseny	28 dies (56h)	1314,40€
Desenvolupament i Proves	35 dies (70h)	1389,20€
Lliurament TFC	21 dies (42h)	987,40€
TOTAL	102 dies (204h)	4638,50€

S'ha de tenir present que en aquesta valoració no s'han tingut en compte les hores destinades al Marge de risc, per tant, si s'haguessin de fer servir, el preu total es veuria incrementant segons les hores utilitzades i el recurs que les hagi fet servir.

La següent taula mostra els preus de les hores del Marge de risc, el temps i els recursos utilitzats:

TASCA	DURACIÓ	RECURS ASSIGNAT	PREU
Planificació			
Marge de risc	1 dia (2h)	Cap de Projecte[75%];Analista[25%]	54,50€
Anàlisi i Disseny			
Marge de risc	2 dies (4h)	Analista	100,00€
Desenvolupament i Proves			
Marge de risc	3 dies (6h)	Programador[85%];Analista[15%]	135,00€
Lliurament TFC			
Marge de risc	2 dies (4h)	Cap de Projecte[50%];Analista[30%]; Programador[20%]	103,50€
TOTAL	8 dies (16h)		393,00€

Per tant, el preu final del projecte incloent el 100% d'hores de risc seria de **5031,50€** (IVA inclòs).

2. Anàlisi i Disseny

Per dur a terme l'anàlisi dels requeriments, s'ha recollit tota la informació de l'enunciat del TFC exposada i demanada pel client. Tota l'anàlisi s'ha fet tenint present el món real i actual del consum energètic i s'ha fet un estudi de les factures, comptadors, consums, lectures, etc. que pot tenir un usuari particular.

2.1. Anàlisi dels requeriments

Els requisits funcionals que s'extreuen de l'enunciat del TFC juntament amb els definits amb el client es detallen a continuació:

- R1: s' ha de poder emmagatzemar les dades d'on es troba físicament un comptador: adreça, localitat, codi postal (CP) i país.
- R2: s' enregistrarà la potència contractada per un client mesurada en KW.
- R3: Es farà una lectura el dia 30 de cada mes i s'emmagatzemarà aquesta dada.
- R4: Haurà dos tipus de lectures de comptadors: presencial o telemàtica
- R5: Es mantindrà un històric del consum mensual de cada comptador.
- R6: S'emmagatzemaran els diferents models de comptadors, així com la empresa on s'ha fabricat, l'any, el país de fabricació, l'adreça i el telèfon de la fàbrica dels comptadors.
- R7: De les centraletes de distribució que connecten als comptadors s'emmagatzemarà: el màxim d'energia elèctrica que poden subministrar a un comptador (Kwh) i l'adreça on es troben situades juntament amb la seva localitat, codi postal i el país.
- R8: S' ha de poder emmagatzemar les dades de les centrals de producció: adreça, localitat, codi postal i país.
- R9: El model ha de permetre saber quantes centrals de distribució alimenta una central de producció.
- R10: Es mantindrà un històric de l'energia elèctrica produïda mensualment mesurat en Kwh per una central de producció.
- R11: S'enregistrarà l'energia màxima produïda per una central de producció i s'emmagatzemarà el seu històric.
- R12: Es mantindrà un històric de les inspeccions de les centrals de producció.
- R13: Les centrals de producció són de diferents tipus: Nuclear, Tèrmica, Carbó, Eòlica i Solar. De cada una d'elles el sistema emmagatzemarà:
 - Nuclear: energia mínima per funcionar i kilograms de rebuig radioactiu generat.
 - Tèrmica: kilograms d'emission de CO2.
 - Carbó: Kilograms d'emissions de CO2.
 - Eòlica: Nombre de molins de vent instal·lats.
 - Solar: Nombre de panells necessaris pel seu funcionament.
- R14: Es calcularà el valor absolut del rebuig o les emissions de tota la història de vida de les centrals de producció.
- R15: De les línies de comunicació que connecten les centrals de producció amb les de distribució, emmagatzemarem la seva capacitat màxima de transport.

- R16: s'enregistraran totes les dades referents a un client: NIF, adreça, localitat, codi postal, país i telèfon.
- R17: La base de dades tindrà com a mínim les següents funcionalitats:
 - Alta, baixa i modificació de clients.
 - Alta, baixa i modificació de comptadors.
 - Alta, baixa i modificació de centraletes de distribució.
 - Alta, baixa i modificació de línies de comunicació.
 - Alta, baixa i modificació de centrals de producció.

Juntament amb la seva informació associada: NIF, adreces, poblacions, consums, potències contractades, kilograms d'emissions de CO₂, kilograms de rebuig radioactiu, nombre de molins, nombre de panells, etc.

- R18: Cada vegada que s'executi un procediment, quedarà registrat en una taula de Logs on s'emmagatzemarà el nom del procediment, la data, els paràmetres d'entrada i els paràmetres de sortida.
- R19: La base de dades ha de permetre els següents procediments de consulta:
 - Un llistat de tots els comptadors on el consum mensual de la data (que es passarà com a paràmetre) superi el 80% de consum mitjà de tots els comptadors de la ciutat (passada com a paràmetre) en aquell mateix període de temps.

El llistat haurà de retornar com a mínim:

- Codi de contracte.
- Potència màxima contractada.
- El tant per cent de consum elèctric consumit en relació al consum mitjà.

Tota aquesta informació es presentarà ordenada de manera ascendent pel tant per cent de consum elèctric consumit.

- Un llistat de les 10 centraletes de distribució que distribueixen més energia.

El llista haurà de retornar:

- L'adreça de la central de distribució.
- L'energia emesa per la centraleta de distribució (valors anuals)
- Màxima energia que permet la centraleta de distribució (valors anuals)

Tota aquesta informació es presentarà ordenada de forma descendent pel valor de l'energia emesa en valor absolut.

- Un llistat de les 10 línies de comunicació més carregades en relació a la seva pròpia capacitat màxima.

El llista haurà de retornar:

- Codi d'identificació.
- Càrrega de la línia, entesa com la potència que suporta la línia entre la capacitat màxima de la línia.
- Energia a la que es pot ampliar la línia tenint en compte les centrals de producció a les que està connectada.

Tota aquesta informació es presentarà de forma descendent per la càrrega de la línia en valor absolut.

- Un llistat dels clients que disposen de comptadors amb servei d'alta disponibilitat tan de centrals de distribució, com de línies de comunicació i de centrals de producció.

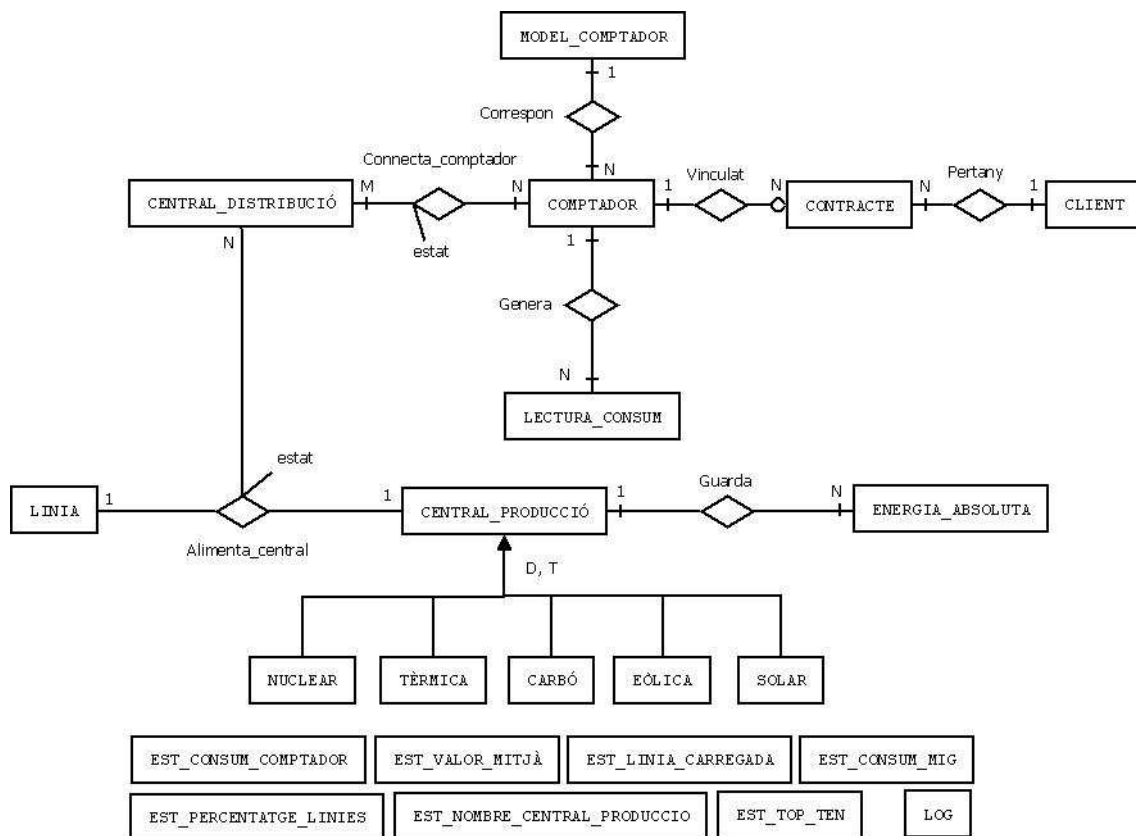
El llista haurà de retornar:

- El DNI del client.
 - El codi de contracte.
 - El model del comptador.
- El consum produït pels comptadors que depenen d'una central (pasada per paràmetre) i l'energia produïda per la central en un període de temps concret (passat per paràmetre).
 - El percentatge de les lectures de comptadors que s'han fet manera presencial i de manera telemàtica en un període de temps concret.
 - Un llistat de comptadors que tinguin un determinat nombre d'anys d'antiguitat.
- R20: Es crearà un model estadístic que oferirà consulta d'informes de dades en temps constant. Les consultes implemetades seràn les següents:
- Donada una central de producció, el consum dels comptadors que depenen de la central.
 - Donada una línia de comunicació i un any concret, el valor mitjà de l'energia consumida.
 - La línia que ha estat més carregada a nivel d'energia consumida.
 - Donat un any concret: percentatge de línies que superen el 50% d'energia consumida.
 - Donat un any concret: el nombre de centrals de producció que generen menys del 30% de producció.
 - Top-10 de comptadors que històricament han tingut més consum.
 - Consum mig de tots els clients.

2.2. Disseny del Model E/R

El següent disseny representa l'estructura de la informació referent a la Base de Dades del Sistema de Control Energètic. Per dur a terme aquest disseny s'ha fet servir un model de dades d'alt nivell "Model Entitat Relació".

El resultat obtingut a partir de l'enunciat com de l'informació intercanviada amb el client és el següent:



2.2.1. Explicació del Model E/R

Tots els comptadors tenen un model de comptador associat. A més, un comptador pot estar vinculat (o no) a un contracte i aquest pertany a un client.

Un client pot tenir diversos contractes de diferents comptadors. Del client tindrem emmagatzemades totes les seves dades personals, juntament amb un codi de client que l'identificarà. Si un client es dona de baixa de la companyia elèctrica, el sistema farà una baixa lògica del seu registre i també del seu contracte associat.

Sabrem si un contracte ha finalitzat per l'atribut de la seva entitat *data_fi*. De la mateixa manera sabrem si un client està actiu o no per la *data_fi* del seu contracte; si existeix la *data_fi* del contracte el client estarà inactiu i si no existeix l'atribut *data_fi*, vol dir que el contracte encara està vigent, per tant, l'estat del registre client estarà actiu a la base de dades.

Pot ser que no s'estigui fent ús dels comptadors per diversos motius: s'ha avariat, encara no s'ha instal·lat, etc. però igualment els tenim registrats a la base de dades a través del seu codi de comptador.

En un contracte tindrem la informació del comptador i del client que en fa ús així com les dates en que s'ha donat d'alta i de baixa d'aquest contracte i la potència contractada.

L'Entitat *comptador* tindrà un atribut *estat* que ens indicarà si el registre del comptador està actiu o no. Estarà actiu quan tingui un contracte associat a un client i inactiu quan no tingui cap contracte associat. De cada comptador tindrem emmagatzemat l'últim consum mensual i el consum absolut, és a dir, el consum que ha tingut un comptador desde que es va realitzar un contracte. A més, es registrarà la seva darrera lectura i tota la informació associada a la seva ubicació física.

Tots els comptadors tenen uns consums que es calcularan a partir de les lectures efectuades en un mes. Per calcular aquesta dada s'ha creat l'entitat *lectura_consum* on tindrem les lectures efectuades juntament amb la seva data i un codi autonumèric (*cod_lect_consum*) que l'identificarà. D'aquestes dades extreurem el consum mensual que el tindrem com atribut a l'entitat *lectura_consum* amb el nom de *consum_mes*.

Les lectures dels comptadors es podran fer de dues modalitats: telemàtiques o presencials.

Les centraletes de distribució són les encarregades de suministrar l'energia als comptadors. Davant la possibilitat que un comptador quedi sense energia, cada comptador estarà connectat a diverses centraletes de distribució per garantir que tots rebin el 100% de l'energia i sempre de manera constant.

D'aquesta rel·lació binària central de distribució i comptador (M:N), sorgeix la interrelació *connecta_comptador* que indicarà que un comptador estarà connectat a diverses centraletes de distribució i, alhora, una centraleta de distribució estarà connectada a diversos comptadors.

Les centraletes de distribució estan alimentades per les centrals de producció a través de les línies de comunicació. D'aquí sorgeix una relació ternària entre les entitats *central_distribució*, *línia* i *central_producció* N:1:1. Una línia pot alimentar a varies centrals de distribució però només connecten a una central de producció.

Les línies de comunicació tenen una capacitat màxima de transport d'energia per suministrar a les centrals de distribució. Aquestes línies són conmutables, és a dir, si una línia cau, la central de distribució no es quedarà sense rebre energia de la central de producció sino que una línia secundària farà les funcions de la primària. D'aquesta manera una línia sempre transporta el 100% de l'energia a la central de distribució. Sempre existirà una línia de comunicació entre les centrals de distribució i les centrals de producció.

Les centrals de producció generen tota l'energia que viatja a través de les línies de comunicació fins les centrals de distribució que s'encarreguen de fer la distribució als comptadors dels diferents clients.

Per saber quanta energia elèctrica produeix mensualment una central de producció, s'ha creat l'entitat *energía_absoluta* que tindrà un històric de tots els consums mensuals de les diferents centrals de producció. Aquesta entitat tindrà un codi autonumèric (*cod_ene_absoluta*) que l'identificarà.

Existeixen centrals de producció de diferents tipus: nuclear, tèrmica, carbó, eòlica i solar (per aquest TFC només existirán aquests cinc tipus de centrals de producció). Com que l'energia que arriba a les centraletes de distribució pot ser generada per un tipus de central de producció, per no crear una entitat de central de producció per cada tipus, s'ha optat per una generalització de la superclasse central de producció que té com a atribut comú el codi de la central de producció, que ens servirà per saber quin tipus de central de producció està generant l'energia.

Tot element d'una central de producció ha de pertànyer a alguna de les seves subclasses, però no a diverses al mateix temps, per això la generalització és Total i Disjunta.

En el model E/R també s'ha tingut en compte les entitats d'estadístiques per poder realitzar les consultes en temps constant i l'entitat Log per poder emmagatzemar les crides als procediments.

2.2.2. Consideracions no reflectides al diagrama

- ✓ Una central de distribució suministra el 100% de l'energia a un comptador, però si aquesta cau es produeix una commutació cap a altre punt d'alimentació. D'aquesta manera la central de distribució que alimenta al comptador sempre proporcionarà el 100% de l'energia.
- ✓ No es podran fer dues lectures en un mateix dia.
- ✓ Es podran fer diverses lectures en un mateix mes.
- ✓ Un client i un comptador es relacionen per un contracte. En el cas que aquesta relació deixi d'existir, el comptador pot tenir un nou contracte amb un altre client (o el mateix) i a la inversa. Aquesta decisió s'ha pres juntament amb el client.
- ✓ Es guardarà el consum (Kwh) de cada comptador de manera mensual.
- ✓ En un mateix mes les lectures es poden fer de manera telemàtica o presencial.
- ✓ Quan un comptador no tingui un contracte d'un client associat (perquè s'ha pogut donar de baixa) es considerarà que està inactiu fins que no torni a tenir altre contracte associat.
- ✓ L'Energia emesa per la central de distribució és la que consumeixen els comptadors, per tant, la que sol·licitaran a la central de producció. (Deixarem de banda l'excés d'energia de les centrals de producció que poden cremar amb resistències internes)

- ✓ No es pot donar d'alta un mateix contracte (codi_contracte) dues vegades encara que sigui pel mateix client. Per exemple, si un client es dona de baixa i passat un temps vol tornar, se li farà un contracte nou (codi_contracte diferent).

2.2.3. Atributs de les entitats

En aquest apartat es detallen els atributs de les entitats del disseny del model E/R. Per identificar la clau primària de les entitats subratllarem l'atribut corresponent.

MODEL_COMPTADOR

cod_model, data_fab, empresa_fab, referència, descripció, adreça, localitat, CP, país, telefon

COMPTADOR

cod_comptador, referència, data_instal, consum_absolut, adreça, localitat, CP, país, estat

CONTRACTE

cod_contracte, data_alta, data_fi, potencia, estat

CLIENT

cod_client, NIF, nom, cognoms, adreça, localitat, CP, país, telefon, estat

LECTURA_CONSUM

cod_lect_consum, data_lectura, consum_mes, lectura, modalitat, estat

ENERGIA_ABSOLUTA

cod_ene_absoluta, data_prod, energia_prod, valid

LÍNIA

cod_línia, capacitat_maxima, estat

CENTRAL_DISTRIBUCIÓ

cod_c_distribucio, maxim_energia, adreça, localitat, cp, país, estat

CENTRAL_PRODUCCIO

cod_c_produccio, energia_maxima, data_inspeccio, capacitat_maxima, adreça localitat, CP, país, estat

NUCLEAR

energia_minima, rebuig

TÈRMICA

emissioCO2

CARBÓ

emissio_CO2

EÒLICA

num_molins

SOLAR

num_panells

EST_CONSUM_COMPTADOR

cod_c_produccio, consum

EST_VALOR_MITJÀ

cod línia, any, valor_mitja

EST_LÍNIA_CARREGADA

cod línia, carrega

EST_PERCENTATGE_LÍNIES

any, percentatge_línies

EST_NOMBRE_CENTRAL_PRODUCCIO

any, num_central_produccio

EST_TOP_TEN

cod comptador, consum

EST_CONSUM_MIG

NIF, cod_client, consum_mig

LOG

cod_log, data, procediment, param_in, param_out, resultat

2.3. Disseny del Model Lògic

El Disseny del Model Lògic pren com a referència el model E/R per a transformar-lo en un conjunt de relacions i claus a partir d'unes regles que s'apliquen sobre les entitats i relacions.

Per realitzar el disseny del model lògic s'ha de tenir en compte la següent notació:

- Les claus primàries apareixen subrallades.
- Les claus foranes es mostren entre claus i s'indica a quina taula apunten.
- Les claus nul·les (que poden estar buides) es mostren amb un asterisc (*).
- Les claus úniques apareixen subratllades amb punts suspensius.

2.3.1. Model Relacional

MODEL_COMPTADOR (cod_model, data_fab, empresa_fab, referència, descripció, adreça, localitat, CP, país, telefon)

COMPTADOR (cod_comptador, referència, cod_model, data_instal*, consum_absolut*, estat, adreça, localitat, CP, país)

on {cod_model} referència MODEL_COMPTADOR

CONTRACTE (cod_contracte, cod_comptador, cod_client, data_alta, data_fi*, potencia, estat)

on {cod_comptador} referència COMPTADOR,

i {cod_client} referència CLIENT

CLIENT (cod_client, NIF, nom, cognoms, adreça, localitat, CP, país, telefon, estat)

LECTURA_CONSUM (cod_lect_consum, cod_comptador, data_lectura, consum_mes, lectura, modalitat, estat)

on {cod_comptador} referència COMPTADOR

CONNECTA_COMPTADOR (cod_comptador, cod_c_distribucio, estat)

on {cod_comptador} referència COMPTADOR

i {cod_c_distribucio} referència CENTRAL_DISTRIBUCIO

ENERGIA_ABSOLUTA (cod_ene_absoluta, cod_c_produccio, data_prod, energia_prod, valid)

on {cod_c_produccio} referència CENTRAL_PRODUCCIO

LÍNIA (cod_linia, capacitat_màxima, estat)

CENTRAL_DISTRIBUCIÓ (cod_c_distribucio, maxim_energia, adreça, localitat, cp, país, estat)

ALIMENTA_CENTRAL (cod_c_distribucio, cod_linia, cod_c_produccio, estat)

on {cod_c_distribucio} referència CENTRAL_DISTRIBUCIO,

{cod_linia} referència LÍNIA

i {cod_c_produccio} referència CENTRAL_PRODUCCIO

CENTRAL_PRODUCCIÓ (cod_c_produccio, energia_maxima, data_inspeccio, adreça, localitat, CP, país, estat)

NUCLEAR (cod_c_produccio, energia_minima, rebuig)

on {cod_c_produccio} referència CENTRAL_PRODUCCIO

TÈRMICA (cod_c_produccio, emissioCO2)

on {cod_c_produccio} referència CENTRAL_PRODUCCIO

CARBÓ (cod_c_produccio, emissio_CO2)

on {cod_c_produccio} referència CENTRAL_PRODUCCIO

EÒLICA (cod_c_produccio, num_molins)

on {cod_c_produccio} referència CENTRAL_PRODUCCIO

SOLAR (cod_c_produccio, num_panells)

on {cod_c_produccio} referència CENTRAL_PRODUCICIO

EST_CONSUM_COMPTADOR (cod_c_produccio, consum)

on {cod_c_produccio} referència CENTRAL_PRODUCICIO

EST_VALOR_MITJÀ (cod_línia, any, valor_mitja)

on {cod_línia} referència LÍNIA

EST_LÍNIA_CARREGADA (cod_línia, carrega)

on {cod_línia} referència LÍNIA

EST_PERCENTATGE_LÍNIES (anyo, percentatge_línies)

EST_NOMBRE_CENTRAL_PRODUCICIO (anyo, num_central_produccio)

EST_TOP_TEN (cod_comptador, consum)

on {cod_comptadors} referència COMPTADOR

EST_CONSUM_MIG (cod_client, NIF, consum_mig)

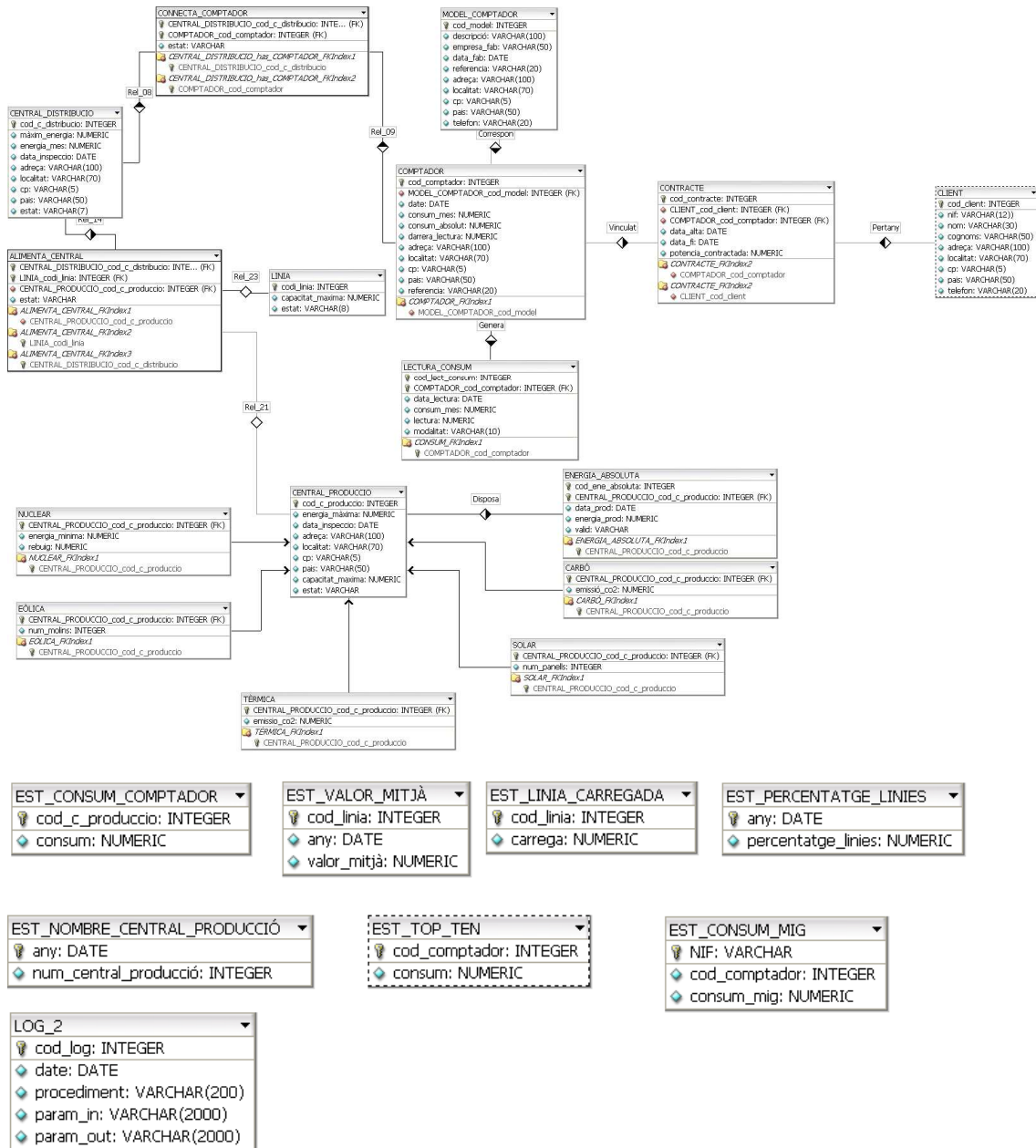
on {cod_client} referència CLIENT

LOG (cod_log, data, procediment, param_in, param_out, resultat)

2.3.2. Restriccions d'integritat referencial

- Si es dona de baixa un comptador de l'entitat *comptador*, s'esborraran en cascada tots els consums i lectures associades de l'entitat *lectura_comptador*.
- Si es dona de baixa un client, el comptador associat passarà a estar inactiu fins que es realitzi un nou contracte per aquest comptador.
- No es pot esborrar un contracte si existeix un comptador i un client associat.
- Si es dona de baixa una central de producció perquè deixa d'existir s'esborraran en cascada l'històric de consums associats.
- No es pot esborrar una central de distribució associada a un comptador.
- No es pot esborrar una línia que comuniqui una central de distribució amb una de producció.
- El consum dels comptadors, centrals de distribució i centrals de producció no podrà tenir valors negatius.

2.3.3. Diagrama Lògic Relacional



2.4. Disseny Físic

Una vegada s'ha creat el disseny lògic de la base de dades, el següent pas és elaborar un disseny físic. Això és el que s'explica en els següents apartats.

2.4.1. Creació de la Base de Dades

Per la creació de la Base de Dades donarem per suposat que on es presenti el model de dades ja haurà un servidor en funcionament. El paquet Sistema Gestor de Base de Dades (SGBD) també estarà instal·lat i la base de dades on s'implementarà el nou model ja estarà creada. Per tant, només serà necessari crear els objectes referents al disseny d'aquest TFC.

2.4.2. Tablespaces

La base de dades tindrà dues unitats lògiques de tablespaces: una per les dades i una altre pels índex. D'aquesta manera, les taules de dades s'emmagatzemaran als tablespaces de dades i els índex en els tablespaces d'índex.

El tablespaces de dades es diu ENERGIA_DAT i el dels index ENERGIA_IND.

En l'script *1_Tablespaces.sql* es troba el codi corresponent.

2.4.3. Usuaris

En un principi el client no ha demanat cap tipus d'usuari especial ni amb cap restricció a nivell d'accés a la base de dades, tot i així, s'ha pensat crear un usuari amb el nom USR_ENE creat per l'usuari SYSTEM que podrà accedir a les diferents taules i funcionalitats de l'aplicació.

Aquest usuari serà propietari de tots els objectes del model de la base de dades i tindrà privilegis per actuar a nivell de tablespaces, creació d'objectes, etc.

En l'script *2_Creació_Usuari.sql* es troba el codi corresponent.

2.4.4. Taules

En aquest apartat s'explica de manera breu les taules que s'han donat d'alta a la base de dades a partir del disseny realitzat. El detall de les taules es troba en l'annex: `..\taules\Taules.docx`

- **Taules principals**

- MODEL_COMPTADOR: conté tots els models d'un comptador.
- COMPTADOR: conté les dades referents als comptadors.

- CONTRACTE: conté les dades referents al contracte que té un client d'un comptador.
 - CLIENT: conté les dades referents als clients consumidors d'energia elèctrica.
 - CENTRAL_DISTRIBUCIO: conté les dades referents a les centrals de distribució.
 - LÍNIA: conté les dades referents a les línies que connecten les centrals de producció amb les de distribució.
 - CENTRAL_PRODUCICIO: conté les dades referents a les centrals de producció.
 - NUCLEAR: conté les dades referents a les centrals de producció de tipus nuclear.
 - TERMICA: les dades referents a les centrals de producció de tipus tèrmica.
 - CARBO: conté les dades referents a les centrals de producció de tipus carbó.
 - EOLICA: conté les dades referents a les centrals de producció de tipus eòlica.
 - SOLAR: conté les dades referents a les centrals de producció de tipus solar.
 - LOG: conté les dades referents als logs de la base de dades.
- **Taules intermitjes**
 - CONNECTA_COMPTADOR: mostra la rel·lació entre els comptadors i les centrals de distribució.
 - ALIMENTA_CENTRAL: mostra la rel·lació entre les centrals de producció, línies i centrals de distribució.
- **Taules auxiliars**
 - LECTURA_CONSUM: conté les dades referents a les lectures dels comptadors juntament amb els seus consums.
 - ENERGIA_ABSOLUTA: conté els registres de l'energia produïda mensualment per una central de producció.
- **Taules d'estadístiques**
 - EST_CONSUM_COMPTADOR: retorna el consum del comptador que depèn d'una central de producció.
 - EST_VALOR_MITJA: retorna el valor mitjà de l'energia consumida per una línia de comunicació en un any concret.
 - EST_LÍNIA_CARREGADA: retorna la línia que ha estat més carregada a nivell d'energia consumida.
 - EST_PERCENTATGE_LÍNIES: retorna el percentatge de línies que superen el 50% d'energia consumida en un any concret.
 - EST_NOMBRE_CENTRAL_PRODUCICIO: retorna el nombre de centrals de producció que generen menys del 30% de producció en un any concret.
 - EST_TOP_TEN: retorna els deu comptadors que històricament han tingut més consum.
 - EST_CONSUM_MIG: retorna el consum mig de tots els clients.

2.4.5. Seqüències

Els codis autoincrementals de les diferents taules es faran a partir de seqüències que es troben a l'arxiu *4_seqüències.sql*.

Tot seguit es detallen les seqüències desenvolupades a la base de dades:

- SEQ_COD_LOG: seqüència que genera els codis d'indentificació de la taula LOG
- SEQ_COD_CLIENT: seqüència que genera els codis d'indentificació de la taula CLIENT
- SEQ_COD_MODEL: seqüència que genera els codis d'indentificació de la taula MODEL_COMPTADOR
- SEQ_COD_COMPTADOR: seqüència que genera els codis d'indentificació de la taula COMPTADOR
- SEQ_COD_CONTRACTE: seqüència que genera els codis d'indentificació de la taula CONTRACTE
- SEQ_COD_C_DIST: seqüència que genera els codis d'indentificació de la taula CENTRAL_DISTRIBUCIO
- SEQ_COD_LÍNIA: seqüència que genera els codis d'indentificació de la taula LÍNIA
- SEQ_COD_C_PRO: seqüència que genera els codis d'indentificació de la taula CENTRAL_PRODUCCIO

2.4.6. Triggers

S'han desenvolupat uns disparadors que introduiran els codis autonumerics generats per les seqüències. Aquest es troben dintre del fitxer *4_seqüències.sql*.

Tot seguit es datallen els triggers desenvolupats a la base de dades:

- INS_COD_LOG: al inserir un nou registre a la taula LOG assigna un nou codi correlatiu al número de la seqüència SEQ_COD_LOG.
- INS_COD_CLIENT: al inserir un nou registre a la taula CLIENT assigna un nou codi correlatiu al número de la seqüència SEQ_COD_CLIENT
- INS_COD_MODEL: al inserir un nou registre a la taula MODEL assigna un nou codi correlatiu al número de la seqüència SEQ_COD_MODEL
- INS_COD_COMPTADOR: al inserir un nou registre a la taula COMPTADOR assigna un nou codi correlatiu al número de la seqüència SEQ_COD_COMPTADOR
- INS_COD_CONTRACTE: al inserir un nou registre a la taula CONTRACTE assigna un nou codi correlatiu al número de la seqüència SEQ_COD_COMPTADOR
- INS_COD_C_DIST: al inserir un nou registre a la taula CENTRAL_DISTRIBUCIO assigna un nou codi correlatiu al número de la seqüència SEQ_COD_C_DIST
- INS_COD_LÍNIA: al inserir un nou registre a la taula LÍNIA assigna un nou codi correlatiu al número de la seqüència SEQ_COD_LÍNIA
- INS_COD_C_PRO: al inserir un nou registre a la taula CENTRAL_PRODUCCIO assigna un nou codi correlatiu al número de la seqüència SEQ_COD_C_PRO

3. Implementació

Una vegada creada la base de dades s'ha procedit a fer la implementació del codi per tal de poder realitzar totes les funcionalitats demanades pel client. Aquest codi anirà adjunt aquest document.

En aquest apartat s'expliquen els diferents procediments i funcionalitats.

L'execució de tots els procediments retorna:

- ✓ OK: si s'ha donat d'alta correctament
- ✓ ERROR: tipus d'error si l'alta no s'ha executat com s'esperava

3.1. Procediments d'alta, baixa i modificació

- **p_altaClient**: gestiona l'alta dels clients a la companyia elèctrica. Es faran les següents comprovacions:
 - Es verifica que els camps no nuls no siguin nuls.
 - Verifiquem que el client no existeix a la base de dades.

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_nif	X	varchar2		X	
n_nom	X	varchar2		X	
n_cognoms	X	varchar2		X	
n_adreça	X	varchar2		X	
n_localitat	X	varchar2		X	
n_cp	X	varchar2		X	
n_pais	X	varchar2		X	
n_telefon	X	varchar2		X	
n_estat	X	varchar2		X	

Tipus d'error:

- ERROR: El client ja existeix
- ERROR: Hi ha camps buits que no poden ser nuls

Exemple d'ús:

```
p_altaClient ('12345678A', 'Marta', 'Perez Gomez', 'Carrer Aribau',
'Barcelona', '08960', 'Spain', 934532123, 'actiu');
```

- **p_modClient**: gestiona la modificació dels clients dintre de la base de dades.
 - Es verifica que els camps no nuls no siguin nuls.
 - Es verifica que el client a modificar existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_nif	X	varchar2		X	
n_nom	X	varchar2		X	
n_cognoms	X	varchar2		X	
n_adreça	X	varchar2		X	
n_localitat	X	varchar2		X	
n_cp	X	varchar2		X	
n_pais	X	varchar2		X	
n_telefon	X	varchar2		X	

Tipus d'error:

- ERROR: El codi del client no existeix
- ERROR: Existeixen camps buits que no poden ser nuls

Exemple d'ús:

```
p_modClient (1, '12341234M', 'Joan', 'Humet Bañon', 'Carcereny 44', 'Sant Joan Despi', 08970, 'Spain', 937483748, 'actiu');
```

- **p_baixaClient:** gestiona la baixa lògica dels clients. Si el registre d'un client es posa com inactiu, el contracte associat també quedarà com inactiu.
 - Es verifica que el codi client existeixi.
 - No es podrà activar un client inactiu. En aquest cas s'haurà de fer una alta nova.
 - El codi del client ha d' existir.

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_client	X	integer		X	
n_estat	X	varchar2		X	

Tipus d'error:

- ERROR: El codi del client no existeix
- ERROR: El codi del client no pot ser nul
- ERROR: No es pot activar un client inactiu

Exemple d'ús:

```
exec p_baixaClient (2, 'inactiu');
```

- **p_altaContracte:** gestiona l'alta dels contractes.
 - Es verifica que el codi_comptador existeix a la taula COMPTADOR.
 - Es verifica que el codi_client existeix a la taula CLIENT
 - Es verifica que la data d'alta existeix (no pot ser superior a la data del sistema)

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_cod_client	X		integer	X	
n_data_alta	X		date	X	
n_data_fi		X	date	X	
n_potencia	X		number	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: No existeix el codi del comptador
- ERROR: No existeix el codi del client
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La data_alta introduïda no pot ser superior a la data actual

Exemple d'ús:

```
p_altaContracte (1, 1, '05/02/2010', '', 4600, 'actiu');
```

- **p_baixaContracte:** gestiona la baixa dels contractes de la base de dades.
 - Es verifica que el codi_contracte no sigui nul
 - Es verifica que existeix el codi_contracte

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_contracte	X		integer	X	

Tipus d'error:

- ERROR: El codi del contracte no existeix
- ERROR: El codi del contracte no pot ser nul

Exemple d'ús:

```
p_baixaContracte (1);
```

- **p_modContracte:** gestiona la modificació dels contractes donats d'alta a la base de dades.
 - Es verifica que el contracte a modificar existeix
 - Es verifica que el codi_comptador existeix en la taula COMPTADOR
 - Es verifica que el codi_client existeix en la taula CLIENT
 - Es verifica que la data_alta existeix (no pot ser superior a la data del sistema)
 - Es verifica que la data_fi existeix (no pot ser superior a la data del sistema)
 - Es verifica que la data_fi del contracte és superior a la data_alta

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_contracte	X		integer	X	
n_cod_comptador	X		integer	X	
n_cod_client	X		integer	X	
n_data_alta	X		date	X	
n_data_fi		X	date	X	
n_potencia	X		number	X	

Tipus d'error:

- ERROR: El codi del contracte no existeix
- ERROR: El codi del comptador no existeix
- ERROR: No existeix el codi del client
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La data_alta introduïda no és correcte
- ERROR: La data fi introduïda no és correcte
- ERROR: La data_fi no pot ser inferior a la data_fi del contracte

Exemple d'ús:

```
p_modContracte (3, 2, 2, '01/04/2010', '01/01/2012', 1000);
```

➤ **p_altaComptador**: gestiona l'alta dels comptadors.

- Es verifica que els camps no siguin nuls
- Es verifica que el codi_model existeix a la taula MODEL_COMPTADOR
- Es verifica que no existeix la referència de comptador
- Es verifica que la data existeix (no pot ser superior a la data del sistema)
- Es verifica que el camp estat està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_model	X		integer	X	
n_referència	X		varchar2	X	
n_data_instal	X		date	X	
n_consum_absolut		X	number	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp		X	varchar2	X	
n_pais	X		varchar2	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: No existeix el codi del model
- ERROR: La referència del comptador ja existeix
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La data d'instal·lació introduïda no pot ser superior a la data actual
- ERROR: El camp estat no s'ha escrit correctament

Exemple d'ús:

```
p_altaComptador(1, 'REF001', '05/02/2010', '', 'Carrer Aribau',
'Barcelona', '08960', 'Spain', 'actiu');
```

- **p_baixaComptador**: gestiona les baixes lògiques dels comptadors. Si el registre d'un comptador queda inactiu, el contracte i lectures associades quedaran inactives.
 - Es verifica que codi del comptador no sigui nul
 - Es verifica que el codi del comptador existeix
 - Es verifica que un comptador no passi d'estat inactiu a actiu
 - Es verifica que el camp *estat* estigui escrit correctament

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi del comptador no existeix
- ERROR: El codi del comptador és nul
- ERROR: No es pot activar un comptador inactiu
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
exec p_baixaComptador (1, 'inactiu');
```

- **p_modComptador**: gestiona la modificació dels comptadors donats d'alta a la taula COMPTADOR.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el comptador a modificar existeix
 - Es verifica que el codi_model existeix a la taula MODEL_COMPTADOR

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_cod_model	X		integer	X	
n_referència	X		varchar2	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp		X	varchar2	X	
n_pais	X		varchar2	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi del comptador no existeix
- ERROR: No existeix el codi del model
- ERROR: Existeixen camps buits que no poden ser nuls

Exemple d'ús:

```
p_modComptador (4,3, 'RSP123', 'adreça', 'localitat', 09090,
'spain', 'actiu');
```

- **p_altaModelComptador:** gestiona l'alta dels diferents models de comptadors
 - Es verifica que els camps no siguin nuls
 - Es verifica que no existeix la referència del model de comptador
 - Es verifica que la data existeix (no pot ser superior a la data del sistema)

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_data_fab	X		date	X	
n_referència	X		varchar2	X	
n_empresa_fab	X		varchar2	X	
n_descripcio	X		varchar2	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	
n_telefon	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de model de comptador ja existeix
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La data introduïda no pot ser superior a la data actual

Exemple d'ús:

```
p_altaModelComptador('01/01/1999', 'ES0109', 'Energia Madrid',
'Comptadors de gran potència', 'Castellana 10', 'Madrid', 05632,
'Spain', 912342234);
```

- **p_baixaModelComptador:** gestiona la baixa dels models de comptadors.
 - Es verifica el cod_model no pot ser nul
 - Es verifica que existeix el model a modificar

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_model	X		integer	X	

Tipus d'error:

- ERROR: El codi de model no existeix
- ERROR: Existeixen camps buits que no poden ser nuls

Exemple d'ús:

```
p_baixaModcomptador(1);
```

- **p_modModelComptador**: gestiona la modificació dels models de comptadors.
 - Es verifica el cod_model no pot ser nul
 - Es verifica que no existeix la referència del model de comptador

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_model	X		integer	X	
n_data_fab	X		date	X	
n_referència	X		varchar2	X	
n_empresa_fab	X		varchar2	X	
n_descripcio	X		varchar2	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	
n_telefon	X		varchar2	X	

Tipus d'error:

- ERROR: El codi del model no existeix
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La data introduïda no pot ser superior a la data actual

Exemple d'ús:

```
p_modModelComptador(1, '01/01/1999', 'ES0109', 'Energia Madrid',
'Comptadors de gran potència', 'Castellana 10', 'Madrid', 05632,
'Spain', 912342234);
```

- **p_altaLecturaConsum**: gestiona l'alta de les lectures d'un comptador. En aquest procediment també s'han inclòs crides a procediments d'estadístiques.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_comptador existeix a la taula COMPTADOR
 - Es verifica que el camp 'modalitat' està escrit correctament: telemàtica o presencial
 - Es verifica que el camp 'vàlid' està escrit correctament: si o no
 - Es verifica que no s'introdueixi una lectura que sigui anterior a al data d'alta de contracte
 - Es verifica que no es faci una lectura d'un comptador que estigui com 'inactiu'
 - Es verifica que no es facin dos lectures en el mateix dia

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_data_lectura	X		date	X	
n_consum_mes	X		number	X	
n_lectura	X		number	X	
n_modalitat	X		varchar2	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *modalitat* no s'ha escrit correctament
- ERROR: no es pot fer una lectura d'un comptador inactiu
- ERROR: El camp *valid* no s'ha escrit correctament
- ERROR: no existeix codi_comptador
- ERROR: La data_lectura no pot ser anterior a la data_alta del contracte
- ERROR: No es poden fer dues lectures en el mateix dia d'un mateix comptador

Exemple d'ús:

```
p_altaLecturaConsum (1, '07/02/2010', '', 100, 'presencial', 'actiu');
```

- **p_baixaLecturaConsum:** gestiona la baixa lògica de la o les lectures d'un comptador.
 - Es verifica que el cod_lect_consum no sigui nul
 - Es verifica que existeix el codi de la LECTURA_CONSUM

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_lect_consum	X		integer	X	
n_valid	X		varchar2	X	

Tipus d'error:

- ERROR: El codi lectura_consum no existeix
- ERROR: El codi de lectura_consum no pot ser nul

Exemple d'ús:

```
p_baixaLecturaConsum (2, 'inactiu');
```

- **p_altaConnectaComptador:** gestiona l'alta de les relacions entre comptador i central de distribució.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_comptador existeix a la taula COMPTADOR
 - Es verifica que no existeix el cod_c_distribucio a la taula CENTRAL_DISTRIBUCIO
 - Es verifica que el camp *estat* està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_cod_c_distribucio	X		integer	X	
n_estat			varchar2	X	

Tipus d'error:

- ERROR: No existeix el codi del comptador
- ERROR: No existeix el codi de la central de distribució
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_altaconnectaComptador (1, 1, 'actiu');
```

- **p_baixaConnectaComptador**: gestiona la baixa de les relacions entre comptador i central de distribució.
 - Es verifica que el cod_comptador no sigui nul
 - Es verifica que el cod_c_distribucio no sigui nul
 - Es verifica que el codi del comptador existeixi
 - Es verifica que el codi de la central de distribució existeixi

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_comptador	X		integer	X	
n_cod_c_distribucio	X		integer	X	
n_estat			varchar2	X	

Tipus d'error:

- ERROR: El codi del comptador no existeix
- ERROR: El codi de la central de distribucio no existeix
- ERROR: El codi del comptador és nul
- ERROR: El codi de la central de distribucio és nul

Exemple d'ús:

```
p_baixaConnectaComptador (1, 1, 'inactiu');
```

- **p_altaCentralDistribucio**: gestiona l'alta de les centrals de distribució.
 - Es verifica que que els camps no siguin nuls
 - Es verifica que el camp *estat* està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_maxim_energia	X		number	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: Existeixen camps nuls que no haurien de ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_altaCentralDistribucio (3500, 'Poligon Barcelona', 'Barcelona',
08980, 'Spain', 'actiu');
```

➤ **p_modCentralDistribucio**: gestiona la modificació de les centrals de distribució.

- Es verifica que que els camps no siguin nuls
- Es verifica que el camp 'estat' està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_distribucio	X		integer	X	
n_maxim_energia	X		number	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la central de distribucio no existeix
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_modCentralDistribucio (1,2000, 'Poligon Barcelona', 'Barcelona',
'', 'Spain');
```

➤ **p_baixaCentralDistribucio**: gestiona la baixa de les centrals de distribució.

- Es verifica que el cod_c_distribucio no sigui nul
- Es verifica que el codi de la central de distribució existeixi

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_distribucio	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la central de distriucio no existeix
- ERROR: El codi de la central de distribució no pot ser nul

Exemple d'ús:

```
p_baixaCentralDistribucio (1, 'inactiu');
```

- **p_altaLínia:** gestiona l'alta de les línies
 - Es verifica que els camps no siguin nuls
 - Es verifica que el camp *estat* estigui escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_capacitat_maxima	X		number	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp "estat" no s"ha escrit correctament

Exemple d'ús:

```
p_altaLínia (4000, 'actiu');
```

- **p_modLínia:** gestiona la modificació de les línies
 - Es verifica que els camps no siguin nuls
 - Es verifica que que la línia a modificar existeix
 - Es verifica que el camp *estat* està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_línia	X		integer	X	
n_capacitat_maxima	X		number	X	

Tipus d'error:

- ERROR: El codi de la línia no existeix
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_modLínia (3, 2000);
```

- **p_baixaLínia:** gestiona la baixa de les línies
 - Es verifica que el codi de línia no sigui nul
 - Es verifica que el codi de línia existeixi
 - Es verifica que no es pugui activa una línia inactiva, en aquest cas s'hauria de fer una alta nova

- Es verifica que el camp estat estigui escrit correctament

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_línia	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la línia no existeix
- ERROR: El codi de la línia no pot ser nul
- ERROR: No es pot activa una línia inactiva
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_baixaLínia (1, 'inactiu');
```

- **p_altaAlimentaCentral:** gestiona l'alta de la taula que relaciona centrals de distribució, centrals de producció i línies.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el cod_línia existeix a la taula LÍNIA
 - Es verifica que no existeix el cod_c_distribucio a la taula CENTRAL_DISTRIBUCIO
 - Es verifica que no existeix el cod_c_produccio a la taula CENTRAL_PRODUCCIO
 - Es verifica que el camp *estat* està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_distribucio	X		integer	X	
n_cod_línia	X		integer	X	
n_cod_c_produccio	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: No existeix el codi de la central de distribució
- ERROR: No existeix el codi de la línia
- ERROR: No existeix el codi de central de producció
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
p_altaAlimentaCentral (1, 1, 1, 'actiu');
```

- **p_baixaAlimentaCentral:** gestiona la baixa dels registres de la taula que relaciona centrals de distribució, centrals de producció i línies.

- Es verifica que els camps no siguin nuls
- Es verifica que el codi de la central de distribució existeix
- Es verifica que el codi de la línia ha existeix
- Es verifica que el codi de la central de producció existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_distribucio	X		integer	X	
n_cod_línia	X		integer	X	
n_cod_c_produccio	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la central de distribució no existeix
- ERROR: El codi de la línia no existeix
- ERROR: El codi de la central de produccio no existeix
- ERROR: Els camps buits no poden estar buits

Exemple d'ús:

```
p_baixaAlimentaCentral (1, 2, 3, 'inactiu');
```

➤ **p_altaCentralProduccio:** gestiona l'alta de les centrals de produccio.

- Es verifica que els camps no siguin nuls
- Es verifica que la data_inspeccio existeix (no pot ser superior a la data del sistema)
- Es verifica que el camp *estat* està escrit correctament: actiu o inactiu

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_energia_maxima	X		number	X	
n_data_inspeccio	X		date	X	
n_capacitat_maxima	X		number	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El camp *estat* no s'ha escrit correctament
- ERROR: La data_inspeccio introduïda no pot ser superior a la data actual

Exemple d'ús:

```
p_altaCentralProduccio ('','02/03/2010', 250000, 'Poligon
Catalunya', 'Barcelona', '08970', 'Spain', 'actiu');
```

- **p_modCentralProduccio**: gestiona la modificació de les centrals de producció.
 - Es verifica que la central de produccio a modificar existeix
 - Es vericica que el camp *estat* està escrit correctament: actiu o inactiu
 - Es verifica que la *data_inspeccio* existeix (no pot ser superior a la data del sistema)

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_capacitat_maxima	X		number	X	
n_data_inspeccio	X		date	X	
n_adreça	X		varchar2	X	
n_localitat	X		varchar2	X	
n_cp	X		varchar2	X	
n_pais	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la central de proudcció no existeix
- ERROR: El camp *estat* no s'ha escrit correctament
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La *data_inspeccio* introduïda no és correcte

Exemple d'ús:

```
p_modCentralProduccio (1, '', '02/04/2011', 250000, 'Poligon
Catalunya', 'Barcelona', '08970', 'Spain');
```

- **p_baixaCentralProduccio**: gestiona la baixa lògica de les centrals de producció. Si el registre d'una central de producció queda inactiu, els registres de la taula ENERGIA_ABSOLUTA corresponents a la central de producció canviaran el seu valor de vàlid a invàlid.
 - Es verifica que el *cod_c_produccio* no sigui nul
 - Es verifica que el codi de la central de producció existeixi
 - Es verifica que el registre d'una central de producció no pugui passar d'*estat actiu a inactiu*
 - Es verifica que el camp *estat* estigui escrit correctament

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_estat	X		varchar2	X	

Tipus d'error:

- ERROR: El codi de la central de producció no existeix
- ERROR: El codi de la central de producció no pot ser nul
- ERROR: No es pot activar una central de producció inactiva
- ERROR: El camp *estat* no s'ha escrit correctament

Exemple d'ús:

```
exec p_baixaCentralProduccio (1, 'inactiu');
```

- **p_altaEnergiaAbsoluta**: gestiona l'alta dels registres de producció d'energia de les diferents centrals de producció.
 - Es verifica que el *cod_c_produccio* no sigui nul
 - Es verifica que el codi de la central de producció existeixi
 - No es pot introduir la mateixa *data_prod* pel mateix *codi_c_produccio*
 - Es verifica que el camp *valid* està escrit correctament: si o no
 - Es verifica que els camps no siguin nuls

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_data_prod	X		date	X	
n_energia_prod	X		number	X	
n_valid	X		varchar2	X	

Tipus d'error:

- ERROR: No existeix el codi de la central de produccio
- ERROR: El camp *valid* no s'ha informat correctament
- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: La *data_prod* introduïda no pot ser superior a la data actual
- ERROR: No es poden donar d'alta dos *data_prod* pel matix *cod_c_produccio*

Exemple d'ús:

```
p_altaEnergiaAbsoluta (2, '27/01/2010', 1000, 'si');
```

- **p_baixaEnergiaAbsoluta**: gestiona la baixa lògica dels registres de producció d'energia de les diferents centrals de producció.
 - Es verifica que el *cod_ene_absoluta* no sigui nul
 - Es verifica que el codi de ENERGIA_ABSOLUTA existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_ene_absoluta	X		integer	X	
n_valid	X		varchar2	X	

Tipus d'error:

- ERROR: El codi cod_ene_absoluta no existeix
- ERROR: El codi de cod_ene_absoluta no pot ser nul

Exemple d'ús:

```
p_baixaEnergiaAbsoluta (2, 'no');
```

- **p_altaNuclear**: gestiona l'alta de les centrals nuclears.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_c_produccio existeix a la taula CENTRAL_PRODUCICIO

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_energia_minima	X		number	X	
n_rebuig	X		number		

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_altaNuclear (3, 2000, 350);
```

- **p_modNuclear**: gestiona la modificació de les centrals nuclears.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi a modificar existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_energia_minima	X		number	X	
n_rebuig	X		number		

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_modNuclear (3, 2000, 550);
```

- **p_baixaNuclear:** gestiona la baixa de les centrals nuclears.
 - Es verifica que el cod_c_produccio no sigui nul
 - Es verifica que el codi de la central de producció existeixi

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	

Tipus d'error:

- ERROR: El codi de la central de producció no pot ser nul
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_baixaNuclear (1);
```

- **p_altaTermica:** gestiona l'alta de les centrals tèrmiques.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_c_produccio existeix a la taula CENTRAL_PRODUCCIO

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_emissioco2	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_altaTermica (5, 700);
```

- **p_modTermica:** gestiona la modificació de les centrals tèrmiques.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi de la central tèrmica existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_emissioco2	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_modTermica (5, 890);
```

➤ **p_baixaTermica**: gestiona la baixa de les centrals tèrmiques.

- Es verifica que el cod_c_produccio no sigui nul
- Es verifica que el codi de la central producció existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	

Tipus d'error:

- ERROR: El codi de la central de producció no pot ser nu
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_baixaTermica (1);
```

➤ **p_altaCarbo**: gestiona l'alta de les centrals de carbó.

- Es verifica que els camps no siguin nuls
- Es verifica que el codi_c_produccio existeix a la taula CENTRAL_PRODUCICIO

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_emissio_co2	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_altaCarbo (1, 500);
```

➤ **p_modCarbo**: gestiona la modificació de les centrals de carbó.

- Es verifica que els camps no siguin nuls
- Es verifica que el codi de la central de carbó existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_emissio_co2	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_modCarbo (1, 700);
```

- **p_baixaCarbo**: gestiona la baixa de les centrals de carbó.
 - Es verifica el cod_c_produccio no sigui nul
 - Es verifica que el codi de la central de producció existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
rst	X		varchar2		X

Tipus d'error:

- ERROR: El codi de la central de producció no pot ser nul
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_baixaCarbo (1);
```

- **p_altaEolica**: gestiona l'alta de les centrals eòliques.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_c_produccio existeix a la taula CENTRAL_PRODUCICIO

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_num_molins	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_altaEolica (2, 679);
```

- **p_modEolica**: gestiona la modificació de les centrals eòliques.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi de la central de carbó existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_num_molins	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_modEolica (2, 670);
```

- **p_baixaEolica**: gestiona la baixa de les centrals eoliques.
 - Es verifica que el codi de la central de producció existeix
 - Es verifica que el cod_c_produccio no sigui nul

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	

Tipus d'error:

- ERROR: El codi de la central de producció no pot ser nul
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_baixaEolica (2);
```

- **p_altaSolar**: gestiona l'alta de les centrals solars.
 - Es verifica que els camps no siguin nuls
 - Es verifica que el codi_c_produccio existeix a la taula CENTRAL_PRODUCICIO

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_num_panells	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_altaSolar (4, 320);
```

- **p_modSolar**: gestiona la modificació de les centrals solars.
 - Es verifica que els camps no siguin nuls
 - Es verifica que que el codi de la central solar existeix

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	
n_num_panells	X		number	X	

Tipus d'error:

- ERROR: Existeixen camps buits que no poden ser nuls
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_modSolar (4);
```

- **p_baixaSolar**: gestiona la baixa de les centrals solars.
 - Es verifica que codi de la central de producció existeix
 - Es verifica que el cod_c_produccio no sigui nul

PARÀMETRES					
Nom	Obligatori	Opcional	Tipus	Input	Ouput
n_cod_c_produccio	X		integer	X	

Tipus d'error:

- ERROR: El codi de la central de producció no pot ser nul
- ERROR: El codi_c_produccio no existeix

Exemple d'ús:

```
p_baixaSolar (1);
```

3.2. Consultes

En el següent apartat es detallen els procediments que s'han implementat per fer les consultes a la base de dades.

- **p_comptConsMes80:** aquest procediment fa una consulta que retorna el llistat dels comptadors on el consum mensual d'una data en concret ha superat el 80% del consum mig de tots els comptadors en una ciutat específica.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
n_localitat	X	varchar2	X	
n_data	X	date	X	
cod_contracte		integer		X
potencia		number		X
percentatge		number		X

Totes les dades es mostraran de manera ascendent pel tant per cent de l consum elèctric consumit.

Exemple d'ús:

```
exec p_comptConsMes80 ('Salamanca', 092011);
```

- **p_centMesEnergia:** aquest procediment retorna una llista de les 10 centraletes que distribueixen més energia.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
adreça		varchar2		X
energia emesa		number		X
màxima energia		number		X

Totes les dades es mostraran de forma descendent per l'energia emesa en valor absolut.

Exemple d'ús:

```
exec p_centMesEnergia ();
```

- **p_linMesCarregada:** aquest procediment retorna les 10 línies de comunicació més carregades en relació a la seva pròpia capacitat màxima.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
cod_línia		integer		X
carga		number		X
ampliació		number		X

Totes les dades es mostraran de manera descendent per la càrrega de la línia en valor absolut.

Exemple d'ús:

```
exec p_LinMesCarregada ();
```

- **p_comptAltaDisponibilitat:** aquest procediment retorna un llistat dels clients que disposen de comptadors amb servei en alta disponibilitat tant de centrals de distribució, com de línies de comunicació i de centrals de producció. Entenem "alta disponibilitat" per tots aquells comptadors que estan connectats a més d'una central de distribució.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
nif		varchar2		X
codi_contracte		integer		X
model_comptador		integer		X

Exemple d'ús:

```
exec p_comptAltaDisponibilitat ();
```

- **p_consumProdCompt:** aquest procediment retorna el consum produït pels comptadors que depenen d'una central de producció en concret i l'energia produïda per a central de producció en un cert interval de temps.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
n_cod_produccio	X	integer	X	
n_data_inici	X	date	X	
n_data_fi	X	date	X	
central_produccio		integer		X
data_inici		date		X
data_fi		date		X
consum_compt		number		X
energia_prod		number		X

Exemple d'ús:

```
exec p_consumProdCompt (2, '01/03/2011', '02/05/2011');
```

- **p_percentModalitat:** aquest procediment retorna el percentatge de lectures de comptadors efectuades de forma presencial i de forma telemàtica en un cert període de temps.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
n_data_inici	X	date	X	
n_data_fi	X	date	X	
Telemàtica %		number		X
presencial %		number		X

Exemple d'ús:

```
exec p_percentModalitat('01/01/2011', '01/01/2012');
```

- **p_antiguitatComptador:** aquest procediment retorna un llistat dels comptadors que tinguin un determinat nombre d'any d'antiguitat.

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
n_antiguitat	X	date	X	
cod_comptador		integer		X
any_fabricacio		number		X
antiguitat		number		X

Exemple d'ús:

```
exec p_antiguitatCompt (13);
```

- **p_retornLog:** aquest procediment s'ha implementat per retornar l'últim registre inserit a la taula LOG. Es farà servir en l'script *JocdeProves.sql*

PARÀMETRES				
Nom	Obligatori	Tipus	Input	Ouput
param_out		varchar2		X

Exemple d'ús:

```
p_retornLog ();
```

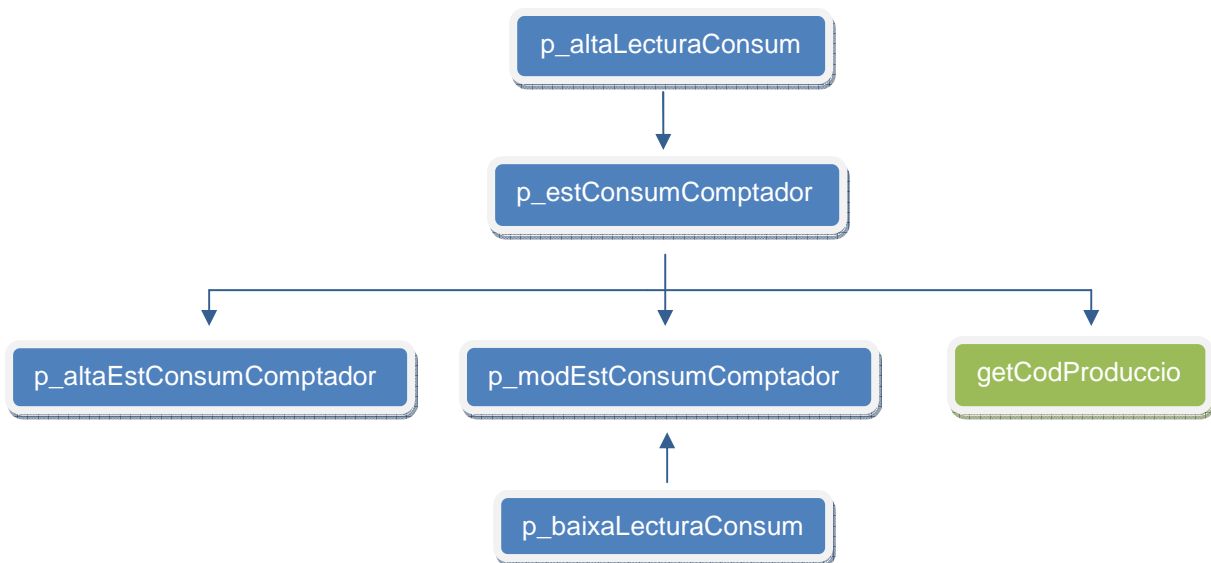
3.3. Estadístiques

En el següent apartat es detallen els procediments i funcions que s'han implementat per resoldre les diferents estadístiques. Aquestes dades es mostraran sempre en temps constant.

- **EST_CONSUM_COMPTADOR**

Aquesta estadística mostra el consum dels comptadors que depenen d'una central de producció en concret.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



- ✓ p_altaLecturaConsum → gestiona l'alta d'una lectura en un comptador.
- ✓ p_estConsumComptador → s'encarrega de fer l'insert o la modificació a la taula EST_CONSUM_COMPTADOR
- ✓ p_altaEstConsumComptador → s'executa quan no existeix el codi de la central de producció a la taula de l'estadística i fa l'insert del cod_produccio i del consum
- ✓ p_modEstConsumComptador → s'executa quan existeix el codi de la central de producció i fa un update del consum.
- ✓ getCodProduccio → funció que retorna el codi d'una central de producció a partir del codi de comptador de la taula lectura_consum.
- ✓ p_baixaLecturaConsum → quan es faci una baixa lògica d'una lectura d'un comptador, obtindrem el codi del comptador, el consum i el codi de la central de producció a partir del codi de lectura. Després es farà l'update a la taula de l'estadística amb el procediment p_modEstConsumComptador

➤ Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

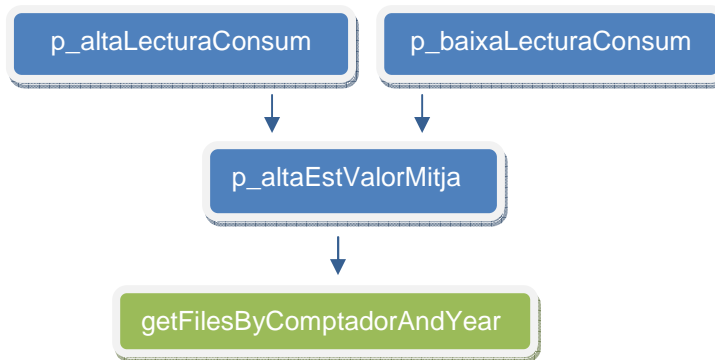
```

SELECT consum
FROM EST_CONSUM_COMPTADOR
WHERE cod_c_produccio = 1
  
```

➤ EST_VALOR_MITJA

Aquesta estadística mostra el valor mitjà de l'energia consumida d'una línia de comunicació en un any concret, tenint en compte que aquest consum depèn dels comptadors que s'alimenten mitjançant aquesta línia.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



- ✓ p_altaLecturaConsum → cada vegada que s'introdueixi una lectura d'un comptador, es cridarà al procediment p_ataValorMitja.
- ✓ p_ataValorMitja → Inserim el nou registre a la taula EST_VALOR_MIJTÀ amb totes les dades ja calculades tenin en compte sempre que la línia introduïda a LECTURA_CONSUM sempre tigi el camp valid = 'si'
- ✓ getFilesByComptadorAndYear → obtenim el número de files a partir d'un comptador i un any. Aquesta dada la farem servir per obtenir la mitja en p_altaValorMitja.
- ✓ p_baixaLecturaConsum → En el moment que es fa una baixa lògica de la taula LECTURA_CONSUM s'actualitza tota la taula EST_VALOR_MITJA

➤ Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

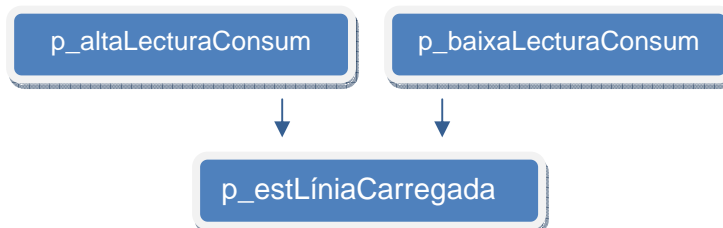
```

SELECT valor_mitja
FROM EST_VALOR_MITJA
WHERE cod_línia = 10 AND anyo = 2011
  
```

• EST_LÍNIA_CARREGADA

Aquesta estadística mostra la línia que ha estat més carregada a nivell d'energia consumida.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



- ✓ p_altaLecturaConsum → cada vegada que s'introdueixi una lectura d'un comptador, es cridarà al procediment p_estLíniaCarregada
- ✓ p_baixaLecturaConsum → quan es faci una baixa lògica de la taula LECTURA_CONSUM es cridarà al procediment p_estLíniaCarregada
- ✓ p_estLíniaCarregada → retorna el codi de la línia més carregada i el valor de la seva càrrega.

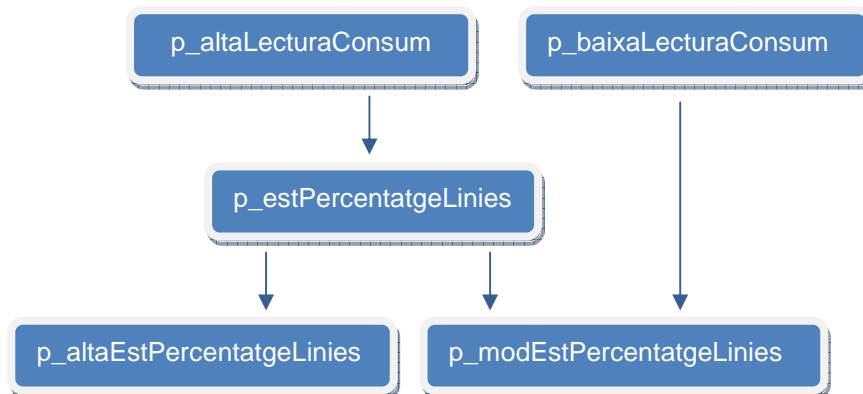
- Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

```
SELECT * FROM EST_LÍNIA_CARREGADA
```

- **EST_PERCENTATGE_LÍNIES**

Aquesta estadística mostra el percentatge de línies que superen el 50% d'energia consumida donat un any concret.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



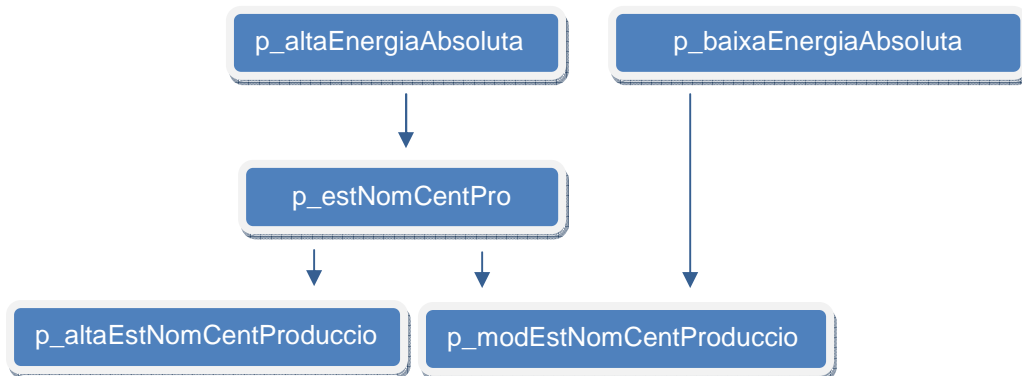
- ✓ p_altaLecturaConsum → cada vegada que s'introdueixi una lectura d'un comptador, es cridarà al procediment p_estPercentatgeLinies
 - ✓ p_estPercentatgeLinies → donat un any concret, calcula el percentatge de les línies i introdueix les dades fent un insert si l'any no existeix a la taula EST_PERCENTATGE_LÍNIES
 - ✓ p_modPercentatgeLinies → si l'any ja existeix a la taula EST_PERCENTATGE_LÍNIES fa un update del percentatge de les línies.
 - ✓ p_baixaLecturaConsum → cada vegada que es faci una baixa lògica d'un registre de la taula LECTURA_CONSUM es cridarà al procediment p_modPercentatgeLinies.
- Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

```
SELECT * FROM EST_PERCENTATGE_LÍNIES
WHERE anyo = 2011
```

- **EST_NOMBRE_CENTRAL_PRODUCCIO**

Aquesta estadística mostra el nombre de centrals de producció que generen menys del 30% de producció donat un any concret.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



- ✓ p_altaEnergiaAbsoluta → cada vegada que s'introdueixi una energia produïda per la central de producció es cridarà al procediment p_estNomCentPro.
- ✓ p_estNomCentPro → en aquest procediment es comprova si existeix l'any a la taula EST_NOMBRE_CENTRAL_PRODUCICIO.
- ✓ p_altaEstNomCentProduccio → es mira el número de centrals que generen menys del 30% de la producció tenint en compte la seva capacitat_maxima (aquesta dada ve donada en valors anuals) i es fa l'insert a la taula
- ✓ p_modEstNomCentProduccio → si ja existeix la central de producció es fa un update de la taula.
- ✓ p_baixaEnergiaAbsoluta → cada vegada que es faci una baixa lògica d'un registre de producció d'energia d'una central de producció es cridarà al procediment p_modEstNomCentProduccio per fer una actualització de la taula de l'estadística.

➤ Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

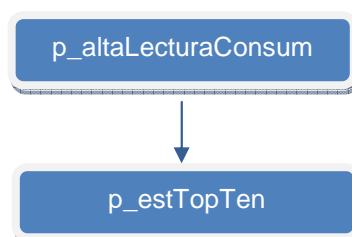
```

SELECT * FROM EST_NOMBRE_CENTRAL_PRODUCICIO
WHERE anyo = 2012
  
```

• EST_TOP_TEN

Aquesta estadística retorna el top-10 dels comptadors que històricament han tingut més consum.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



- ✓ p_altaLecturaConsum → cada vegada que es dona d'alta una lectura d'un comptador a la taula LECTURA_CONSUM es crida al procediment p_estTopTen
- ✓ p_estTopTen → aquest procediment introdueix el top-10 dels comptadors que han tingut més consum desde que es van instal·lar.

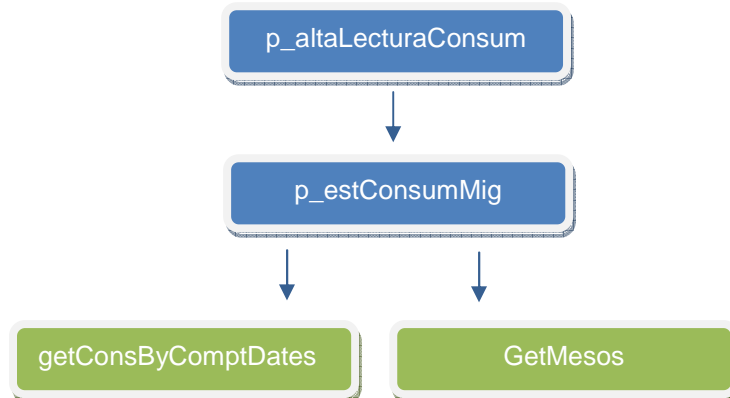
➤ Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

```
SELECT * FROM EST_TOP_TEN
```

- **EST_CONSUM_MIG**

Aquesta estadística retorna el consum mig de tots els clients.

El següent diagrama mostra les crides als diferents procediments i funcions que s'han desenvolupat:



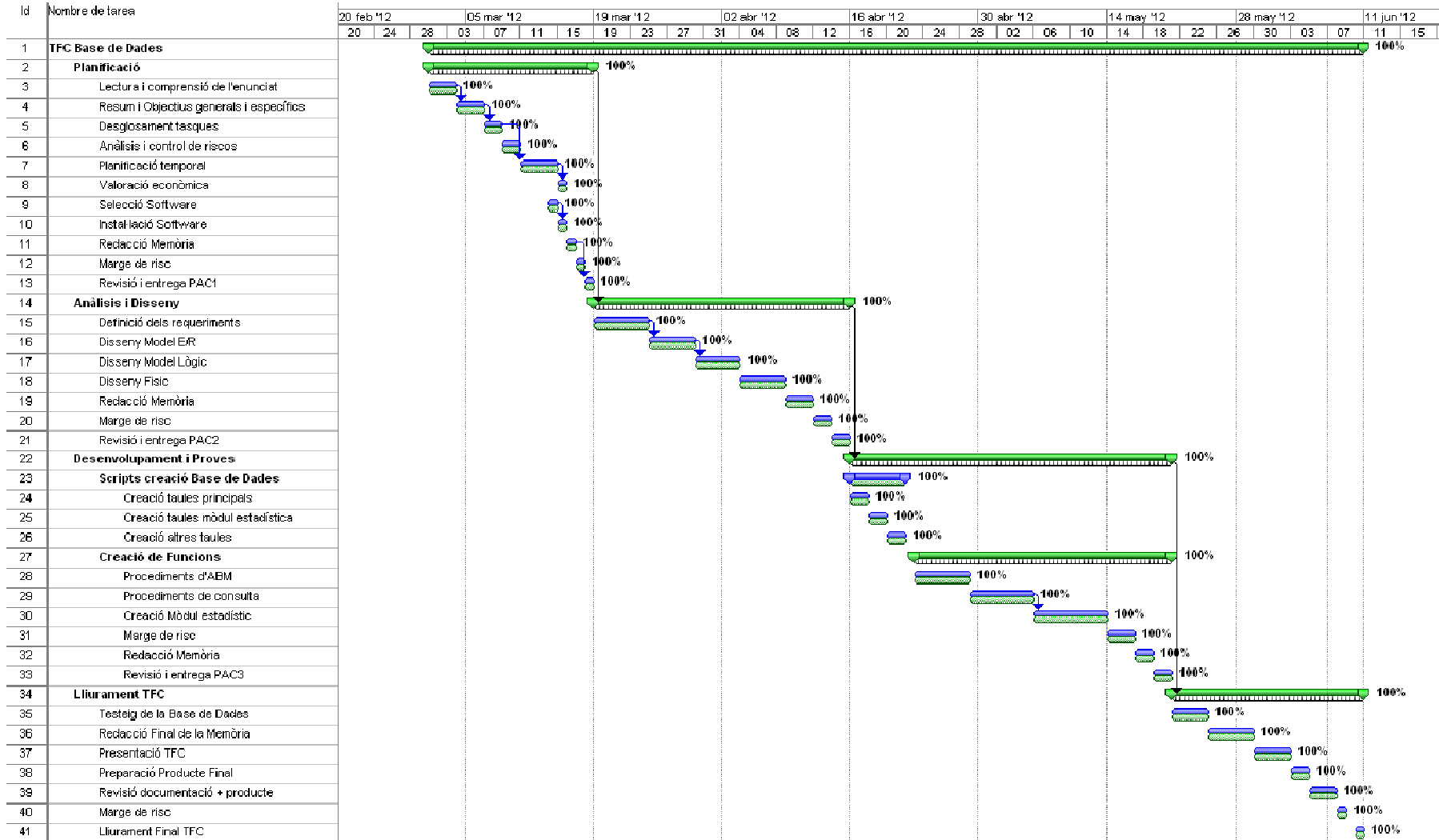
- ✓ p_altaLecturaConsum → cada vegada que es dona d'alta una lectura d'un comptador a la taula LECTURA_CONSUM es crida al procediment p_altaLecturaConsum
- ✓ p_estConsumMig → es divideix el consum total d'un comptador pels mesos que porta consumint des de que es va donar d'alta el seu contracte fins a la seva data fi o fins la data actual si el contracte encara està vigent. Després es fa un insert a la taula EST_CONSUMMIG
- ✓ getConsByComptDates → aquesta funció retorna el consum d'un comptador tant si ha finalitzat el seu contracte com si encara està actiu.
- ✓ getMesos → retorna els números de mesos des de que s'ha donat d'alta un contracte fins a la data actual o bé des de que es va donar d'alta un contracte fins a la seva finalització.

➤ Per obtenir el resultat d'aquesta estadística s'ha d'executar la següent query:

```
SELECT * FROM EST_CONSUM_MIG
```

4. Seguiment i millores del TFC

El següent diagrama de Gantt mostra l'evolució de les tasques planificades a la PAC1:



Del diagrama anterior, s'observa que al llarg de la PAC1, PAC2 i PAC3 s'han desenvolupat les tasques segons la planificació feta a l'inici del TFC. S'ha arribat a les dates claus amb totes les implementacions i memòria corresponents.

També es pot observar que s'han fet servir els temps que es van deixar de '*Marge de risc*'. Això ha sigut causat pels imprevistos trobats al llarg de tot el TFC .

4.1. Possibles millores

En tot el procés del disseny de la base de dades i de l'anàlisi dels requeriments, s'han trobat possibles millores que es podrien fer en un segon evolutiu del projecte.

Aquestes millores es detallen a continuació:

- ✓ Si un mes no hi hagués lectura d'un comptador el següent mes s'estimaria la lectura que manca.
- ✓ Depenent del rang de potència contractada el model de comptador podria ser d'un tipus o altre.
- ✓ Es podria implementar un històric d'inspeccions de centrals de distribucions i de centrals de producció.
- ✓ Ampliar el mòdul estadístic amb el camp ciutat:
 - Quines ciutats són les que consumeixen més energia.
 - En quin període de temps hi ha més consum d'energia en una ciutat determinada.
- ✓ Es podrien crear diferents usuaris amb diferents privilegis per accedir a la base de dades. En aquest projecte s'ha creat l'usuari USER_TFC com a usuari que disposa de tots els privilegis disponibles. Per tal de conservar l'integritat de les dades i de les estructures, seria convenient que només alguns usuaris poguessin realitzar determinades tasques, i que altres que requereixen major coneixement sobre les estructures de dades i taules, només es poguessin realitzar per un nombre limitat i controlat d'usuaris.

5. Joc de proves

S'ha desenvolupat un joc de proves on es demostra el bon funcionament de les funcionalitats implementades així com el control d'errors i de situacions d'excepció de les consultes i de les estadístiques.

Aquestes proves s'han dividit en dues parts:

- Proves de consultes i estadístiques
- Proves d'alta, baixa i modificació

Les proves de consultes i estadístiques es troben en el document: *Joc_de_Proves.docx* que es troba dins de la carpeta *JocDeProves*.

En aquest document, s'han posat totes les crides als procediments i query's que s'han d'executar per tal de fer les consultes i les estadístiques a la base de dades juntament amb una captura del resultat. S'han fet tres tipus de proves:

- ✓ Prova on es mostra el resultat esperat.
- ✓ Prova on es mostra els control d'excepcions.
- ✓ Prova on es mostra que actualitzant registres de les taules de la base de dades, les estadístiques canvien en temps constatat.

Per les proves d'alta, baixa i modificació, ha creat un script *JocdeProves.sql* on es poden executar diferents altes, modificacions i baixes de registres. A més, es poden veure tots els logs de les altes que es fan en el moment de crear la base de dades amb l'script *3_CREACIO_BD.sql* en la taula LOG de la base de dades.

Una vegada s'ha executat el fitxer *JocdeProves.sql* genera un altre en la carpeta *JocDeProves* amb el nom *JocdeProvesMB.txt* on es pot veure el resultat de les execucions dels diferents processos.

! Es recomana executar el fitxer *JocdeProves.sql* després d'haver fet les comprovacions de les estadístiques i de les consultes, ja que així, els resultats mostrats en les captures del document *Joc_de_Proves.pdf* i *JocdeProves.sql* seran els mateixos.

En els processos desenvolupants no s'ha posat el paràmetre 'rst' com a paràmetre de sortida, sino que s'ha implementat com una variable local. Per tant, s'ha hagut de programar un petit procediment (*p_returnLog*) per tal de poder *logar* el resultat de les execucions dels procediments que queden registrats en la taula LOG en el fitxer *JocdeProvesMB.txt*

S'ha tingut en compte a l'hora de programar aquest procediment que tingués el mínim cost possible. En les següents captures es mostren dues maneres d'implementar el procediment i obtenir el mateix resultat, però la primera d'elles obté un cost més petit i és aquesta opció la que s'ha implementat en el projecte.

Menor cost:

The screenshot shows a SQL IDE interface. The query editor contains the following SQL code:

```
SELECT I.DATA, I.PARAM_OUT FROM LOG I
WHERE I.COD_LOG = (SELECT MAX(LL.COD_LOG) FROM LOG LL)
```

Below the query, the 'Explain Plan' window is open, displaying the following execution plan:

Operation	Object Name	Rows	Bytes	Cost	Object Node	In/Out	PStart	PStop
SELECT STATEMENT Optimizer Mode=ALL_ROWS		1		4				
TABLE ACCESS BY INDEX ROWID	USR_ENE.LOG	1	51	2				
INDEX UNIQUE SCAN	USR_ENE.PK_COD_LOG	1		1				
SORT AGGREGATE		1	4					
INDEX FULL SCAN (MIN/MAX)	USR_ENE.PK_COD_LOG	5 K	23 K	2				

Major cost:

The screenshot shows a database query tool interface. At the top, there is a toolbar with various icons. Below the toolbar, the query text is displayed: `SELECT L.DATA, L.PARAM_OUT FROM LOG L WHERE L.DATA = (SELECT MAX(LL.DATA) FROM LOG LL)`. Below the query, the 'Explain Plan' section is visible, showing a table with columns: Operation, Object Name, Rows, Bytes, Cost, Object Node, In/Out, PStart, and PStop. The plan details are as follows:

Operation	Object Name	Rows	Bytes	Cost	Object Node	In/Out	PStart	PStop
SELECT STATEMENT Optimizer Mode=ALL_ROWS		30		61				
TABLE ACCESS FULL	USR_ENE.LOG	30	1 K	30				
SORT AGGREGATE		1	8					
TABLE ACCESS FULL	USR_ENE.LOG	5 K	46 K	30				

6. Conclusió

Pel que s'ha pogut veure al llarg de tot el període de temps que ha durat el desenvolupament del TFC, es poden extreure un sèrie de conclusions a nivell de planificacions de projectes, disseny, desenvolupaments i preu.

En primer lloc, s'ha pogut comprovar la importància de realitzar una bona planificació del projecte a l'inici del treball. Aquesta planificació ha de ser el més realista possible i tenir present els dies d'absència per malatia o vacances per tal que no afecti al desenvolupament de tot el projecte. A més a més, és de vital importància deixar un període de temps per fer les proves de l'aplicatiu, ja que sempre poden sorgir imprevistos d'última hora que requereixin una correcció. Així doncs, és important fer una bona planificació per tal d'arribar a la data d'entrega amb el producte finalitzat i amb garanties de bon funcionament.

S'han determinat unes dates claus per anar entregant documentació al client i que aquest pogués anar revisant i validant, i alhora indicant els errors, imprecisions o divergències a l'hora d'interpretar els diferents punts a desenvolupar. Així s'ha aconseguit corregir els errors el més aviat possible de manera que no afectessin a les etapes posteriors del projecte.

També he observat que la documentació ha de ser molt clara i entenedora pel client, ja que, com que el producte no s'entrega fins al final, és l'única manera de poder demostrar al client que el programador està treballant i està fent una feina ben feta. Això donarà confiança i tranquil·litat al client, fet beneficiós per ambdues parts.

Així doncs, no només és bàsic la fase de l'anàlisi i recollida de requeriments, sinó que també és fonamental implementar un bon disseny ja que del seu resultat dependrà tota la fase de desenvolupament. No voldrà dir que aquesta fase estigui tancada ja que es podrà modificar segons sigui convenient pels errors sorgits posteriorment o per demanda de nous requeriments per part del client.

Pel que respecta al SGBD i l'opció de programar els diferents procediments emmagatzemats, ha esdevingut una eina molt interessant, ja que son els propis procediments els encarregats de fer la validació de dades, comprovar que es compleixen certes condicions, fer crides a altres procediments, etc. Cal destacar que el manteniment del codi és bastant senzill ja que les funcionalitats es troben repartides en diversos procediments i cap d'ells resulta massa llarg o complex.

Finalment, s'ha de valorar el cost del projecte que dependrà del preu per hora del professional contractat (dependrà de l'experiència, empresa contractada, etc). En aquest projecte només ha hagut un professional implicat que ha agafat els rols de diferents professionals segons en l'etapa de desenvolupament que es trobava, i per això en la valoració econòmica s'ha desglosat per perfil professional, d'aquesta manera s'intenta donar un preu de projecte el màxim real possible. A més, s'hauria de tenir en compte altres costos com per exemple les llicències d'Oracle. Arribat a aquest punt, hauria de ser el client qui valors si l'import del projecte i les seves necessitats son rentables per fer la inversió.

7. Glosari

Atribut: Representa una propietat d'una entitat.

BBDD: Base de Dades.

Entitat: Representació d'un objecte o concepte del món real.

Funció: Subprograma que representa un valor que es genera com a resultat de la seva execució.

PAC: Prova d'Avaluació Continuada

PL/SQL: Llenguatge de programació orientat a base de dades sobre Oracle.

Procedure: Subprograma que només indica l'execució d'una seqüència d'instruccions en funció d'uns paràmetres.

Relació: Descriu una interacció entre dos o més entitats.

Seqüència: Procediment emmagatzemat que genera un increment per a obtenir un camp autonumèric.

Clau Forana: Referència a una clau en una altre taula.

Clau Primària: Clau única escollida entre totes les candidates i defineix unívocament a la resta d'atributs de la taula.

Clau Única: Identifiquen de manera única cada registre d'una taula.

Script: Codi Font que executa un procés.

SGBD: Sistema Gestor de Base de Dades

Trigger: subrutina que s'executa de manera automàtica quan es produeix alguna mena de transacció d'una taula.

TFC: Treball Final de Carrera

8. Bibliografia

- **Materials didàctics de les assignatures:** Base de dades 1, Base de dades II, Sistemes de Gestió de base de dades, Estructura de la Informació, Tècniques de Desenvolupament de Software
- **Urman, S(2002).** Oracle 9i Programación PL/SQL. Madrid: McGraw-Hill
- **Oracle Composite Indexes and Foreign Key Constraints:** disponible en web: http://appsdba.com/techinfo/composite_index.pdf
- **Desarrollo de Base de Datos: Casos prácticos desde el análisis a la implementación.** Editorial: Ra-Ma

9. Annexos

Com a documentació complementària a la memòria, es lliura la següent documentació:

- /abm → en aquesta carpeta es troben tots els scripts d'alta, baixa i modificació de registres.
- /consultes → en aquesta carpeta es troben tots els procediments referents a les consultes.
- /estadistiques → en aquesta carpeta es troben tots els procediments i funcions referents a les estadístiques.
- /taules → en aquesta carpeta es troben els scripts de les taules de la base de dades i el document *Taules.pdf* on s'explica la implementació de les taules.
- /JocDeProves → en aquesta carpeta es troba l'script del joc de proves.
- Scripts:
 - 1_Tablespaces.sql → script de creació dels tablespaces.
 - 2_Creació_Usuari.sql → script de creació dels usuaris.
 - 3_CREACIO_BD.sql → script encarregat de cridar als processos encarregats de montar la base de dades. L'execució d'aquest script genera el fitxer *creacioBD.txt* on es comprova que tots els processos s'han executat correctament.
 - 4_tables.sql → script de creació de les taules.
 - 5_seqüències_i_trigger.sql → script de creació de les seqüències i triggers.
 - 6_insertLog.sql → script que insereix els logs dels processos en la taula LOG.
 - 7_estadistiques.sql → script dels processos de les estadístiques.
 - 8_abm.sql → script dels processos d'alta, baixa i modificació.
 - 9_carregaDades.sql → script on es troba totes les altes de registres que es fan a les taules de la base de dades.
 - 10_consultes.sql → script on es troben tots els processos de consulta.
 - DropBD.sql → script que esborra les seqüències, procediments, funcions i taules de la base de dades. L'execució d'aquest script genera el fitxer *DropBD.txt* on es comprova que s'ha donat de baixa la base de dades correctament.
- MANUAL_USUARI.txt → manual on s'explica l'ordre d'execució dels diferents scripts pel correcte funcionament de la base de dades.