

Global Service Desk Database

Sergi Boticario Domínguez

Grau en Enginyeria Informàtica
Base de Dades

Jordi Ferrer Duran

María Isabel Guitart Hormigo

11 de Juny de 2018



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Global Service Desk Database</i>
Nom de l'autor:	<i>Sergi Boticario Domínguez</i>
Nom del consultor:	<i>Jordi Ferrer Duran</i>
Nom del PRA:	<i>María Isabel Guitart Hormigo</i>
Data de lliurament:	<i>06/2018</i>
Titulació o programa:	<i>GEI - Grau en Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>Base de dades</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Base de Dades, Ticket, Relacional.</i>
Resum del Treball	
<p>En aquest treball realitzarem el disseny i implementació d'una base de dades operativa amb funcions de magatzem que serveixi com a suport per a les dades necessàries per a gestionar la creació de peticions i incidències per part dels usuaris finals.</p> <p>Executarem el disseny en 3 etapes: L'etapa conceptual, on plasmarem la realitat del problema en un diagrama UML reflectint les entitats, els seus atributs i les relacions que les uneixen. A continuació l'etapa lògica, on es traduirà el model conceptual a relacions lògiques pròpies d'una base de dades relacionals i finalment l'etapa física, on es traduiran aquestes a sentències SQL per al SGBD triat.</p> <p>A continuació, es generaran tots els procediments necessaris per a gestionar les dades així com aquelles per a efectuar consultes sobre el magatzem.</p> <p>Finalment, es generarà uns exhaustius jocs de proves per a testejar totes les funcionalitats i casos possibles.</p> <p>Els resultats obtinguts han sigut plenament satisfactoris respecte al que el client demanava, i molt acurats respecte a la planificació temporal i assoliment d'objectius considerats a l'inici del treball.</p> <p>L'execució del treball ha servit per posar de manifest la importància d'una correcta planificació prèvia, i també per comprovar que la majoria d'errors es produeixen quan t'enfrontes a una problemàtica de la qual et falta informació.</p>	

Abstract

In this degree project we will design an operational database with warehouse functions that will operate as support for a future application which will handle the creation of requests and incidences by end users.

We will design the database through 3 different stages: The conceptual stage, where we will transfer the problem into an UML diagram, reflecting the entities, their attributes and the relationships between them. The next stage is the logical stage, where the conceptual model will be translated into a collection of logical sentences and finally the physical stage, where these logical sentences will be translated into SQL statements for the chosen DBMS.

Furthermore, we will create all the procedures to manage the data as well as those procedures necessities to generate queries against the data warehouse.

Finally, we will generate an exhaustive test kit to try all the functionalities and make sure that our database is able to cover all the possible eventualities.

As a conclusion, the results obtained were fully satisfactory concerning what the client was asking for, and pretty tight to the schedule that we did at the beginning of the work.

The execution of the project has served to see the importance of an accurate planning, as well as to realize that in the most cases, errors are produced when you have to deal with a problem having a lack of information.

Índex

1. Introducció.....	1
1.1 Context i justificació del Treball	1
1.2 Objectius del Treball.....	1
1.3 Enfocament i mètode seguit.....	2
1.4 Planificació del Treball.....	3
1.4.1 Programari necessari	4
Planificador de tasques	4
Editor de text	4
Editor de diagrames UML	4
Base de dades.....	4
Sistema gestor de base de dades	5
Maquina Virtual.....	5
Editor per a la presentació.....	5
Captura de pantalla i edició de vídeo.....	5
1.4.2 Descomposició en tasques i planificació temporal	6
Feina inicial.....	6
Preparació de l'entorn.....	6
Disseny conceptual de la base de dades	7
Disseny físic de la base de dades	7
Magatzem de dades	8
Prova de la base de dades	8
Documentació a lliurar	9
Presentació en vídeo	9
Avaluació de les competències transversals	10
1.5 Breu sumari de productes obtinguts	11
1.6 Breu descripció dels altres capítols de la memòria	12

2. Disseny Conceptual	13
2.1 Identificació d'entitats, atributs i relacions	13
2.2 Diagrama conceptual.....	15
2.3 Decisions de disseny.....	15
2.3.1 Tipus de tiquet.....	16
2.3.2 Prioritats	17
2.3.3 Usuaris tècnics	18
2.3.4 Registres	19
2.3.5 Enumeracions VS Entitats pròpies	20
2.4 Restriccions textuais.....	21
3. Disseny Lògic	23
3.2 Traducció del model conceptual al model lògic	23
3.3 Normalització.....	26
4. Disseny Físic	32
4.1 Traducció del model lògic a sentències SQL.....	32
5. Procediments.....	33
5.1 País	35
5.2 Seu	35
5.3 Usuari	36
5.4 Interacció.....	37
5.5 NivellSuport	38
5.6 GrupSuport.....	38
5.7 Tècnic.....	39
5.8 Prioritat	40
5.9 GrupServei	41
5.10 Servei	41

5.11 OfertaGrupServei	43
5.12 TipusTicket	43
5.13 Canal	44
5.14 CreacioTicket	44
5.15 Ticket.....	45
5.16 AssigTicketGrup	46
5.17 EstatTicket.....	48
5.18 CanviEstat	48
5.19 EstatSms	49
5.20 Missatge	49
5.21 Log	50

6. Data Warehouse 51

6.1 Anàlisi de les dades requerides.....	51
6.2 Estructura i creació de les taules.....	54
6.3 Procediments d'actualització	57
6.3.1 DW_temps_incidencies_prio.....	58
6.3. DW_petitions_sobre_temps	58
6.3.3 DW_usuari_mes_incidencies	59
6.3.4 DW_top10_petitions_seu	59
6.3.5 DW_tickets_resolts_mes.....	60
6.3.6 DW_servei_petitions.....	60
6.3.7 DW_incidencies_seu_any	61
6.3.8 DW_hores_tickets_L3	62
6.3.9 DW_incidencies_PM	62
6.3.10 DW_sms_no_respostos	62
6.3.11 DW_tickets_tancats.....	63
6.3.12 DW_tickets_per_canal	63
6.4 Procediments de consulta	65

7. Proves	67
7.1 Volum de dades inicial.....	67
7.2 Anàlisi dels casos i funcionalitats	68
7.2.1 Base de dades operativa.....	68
7.2.2 Data Warehouse	72
8. Conclusions.....	76
8.1 Conclusions del treball	76
8.2 Anàlisi sobre l'assoliment d'objectius	77
8.3 Anàlisi del seguiment de la planificació	79
8.4 Línies de treball futur	81
9. Glossari	82
10. Bibliografia.....	83
11. Annexos	i
11.1 Versió prèvia del mapa conceptual	i
11.2 Taules de la base de dades operativa.....	ii
11.3 Procediments d'alta, baixa i modificació de la Base de dades operativa. v	
11.4 Taules del Data Warehouse	xxi
11.5 Procediments d'actualització del Data Warehouse	xxiii
11.6 Procediments de consulta del Data Warehouse.....	xxviii
11.6 Volum de dades inicials.....	xxxi
11.7 Jocs de proves	xxxiii
11.7.1 Base de dades operativa.....	xxxiii
11.7.2 Data Warehouse	xxxv
11.8 Execució dels jocs i resultats.....	xl
11.8.1 Base de dades operativa.....	xl
11.8.2 Data warehouse	xliv

1. Introducció

1.1 Context i justificació del Treball

Ens trobem davant d'una multinacional que treballa en l'àmbit de l'eficiència energètica. Aquesta empresa, amb seus distribuïdes arreu del món, es troba en una fase d'estandardització dels seus processos informàtics i concretament es vol centrar en aquells processos encarregats de donar suport als usuaris.

Actualment, aquests processos d'incidències i peticions es generen de manera estanca en l'àmbit de la seu, i es desitja que el procés d'estandardització aconseguixi un únic sistema global de suport als usuaris que s'ha anomenat Global Service Desk.

Concretament, dins d'aquest servei de suport als usuaris es vol començar per a la definició dels processos de gestió d'incidències i peticions dels serveis informàtics, de manera que l'equip de GSD ha definit els processos que durant a terme l'estandardització d'aquestes tasques en una futura aplicació que gestioni les diferents casuístiques amb les quals els usuaris poden trobar-se.

En efecte, el primer pas per a poder crear el sistema és la implementació d'una base de dades capaç d'emmagatzemar i gestionar les dades que puguin transformar-se en informació útil per a la gestió de les tasques prèviament esmentades.

Així doncs, en aquest treball es vol dissenyar i implementar aquesta base de dades que mantindrà les dades necessàries perquè la futura aplicació pugui donar servei als usuaris finals ubicats a les diferents seus de la multinacional. Aquesta base de dades contindrà tots els mecanismes necessaris per a poder servir la informació així com totes les estructures que calguin per a poder actuar també de magatzem de dades i ser capaç de donar resposta en temps constant a un seguit de consultes preestablertes.

1.2 Objectius del Treball

- ✓ Creació de l'estructura física d'una base de dades relacional capaç d'emmagatzemar les dades necessàries per a generar la informació requerida per una futura aplicació que gestioni els processos d'incidències i peticions.
- ✓ Creació dels procediments necessaris per a donar resposta a les consultes previstes a la base de dades així com les funcionalitats d'alta baixa i modificació.

- ✓ Creació dels mecanismes d'automatització i de millora de rendiment per a la base de dades.
- ✓ Creació de les estructures necessàries per al repositori de dades, fent possible la generació de respostes a consultes predefinides en temps constant.
- ✓ Creació dels scripts necessaris per a la generació de la base de dades de manera automatitzada.
- ✓ Creació d'un joc de proves exhaustiu que cobreixi tots els aspectes de comprovacions a les diferents funcionalitats existents a la base de dades així com el seu rendiment.

1.3 Enfocament i mètode seguit

La finalitat d'aquest projecte serà la creació d'una base de dades nova que compleixi tots els requisits per a donar servei a la futura aplicació que l'empresa vol desenvolupar.

Per a fer possible la creació d'aquesta nova base de dades d'una manera rigorosa i assegurant-nos que aquesta compleix tots els requisits plantejats, seguirem una metodologia de creació dividida en 3 fases clarament diferenciades: disseny conceptual, lògic i finalment físic de la base de dades.

En la primera fase farem un estudi de les dades que la base de dades haurà d'emmagatzemar per a donar resposta a totes les futures consultes i a partir de les quals es podrà emetre informació útil per a l'empresa. Un cop definides aquestes dades, caldrà modelar la realitat en un mapa conceptual que ens servirà com a base per a la futura base de dades. Aquest mapa serà un model ER (entitat-relació)^[1], model conceptual de dades d'alt nivell i independent de la tecnologia que té com a finalitat reflectir en un model conceptual els requisits del món real que siguin d'interès per al problema^[2].

Així doncs, en aquest mapa definirem quines són les entitats, quins atributs tenen cadascuna d'elles (les dades que necessitem desar esmentades prèviament) i finalment la relació que existeix entre cadascuna de les entitats del nostre problema.

En aquest procés intentarem reflectir sobre l'esquema el nombre més gran possible de restriccions que existeixen al món real i que han d'estar presents a la nostra base de dades. Aquelles que no sigui possible d'explicitar de forma gràfica hauran de ser redactades com a restriccions textuais i totes elles s'hauran de tenir en compte per a les futures fases del desenvolupament.

[1] Model ER: https://es.wikipedia.org/wiki/Modelo_entidad-relaci%C3%B3n

[2] Disseny conceptual de bases de dades: PID_00220512. Jordi Casas Roma. Universitat Oberta de Catalunya.

La segona fase es tracta del disseny lògic de la base de dades. En aquesta fase passarem a traduir el diagrama conceptual per a obtenir un esquema lògic de la base de dades. Aquest esquema dependrà del tipus de base de dades que farem servir (en aquest cas una base de dades relacional) però no tindrà en compte les especificacions concretes de la implementació del SGBD^[3]. També en aquesta etapa, farem ús de la teoria de la normalització (basada en la teoria de conjunts, la lògica i l'àlgebra) per a guiar-nos sobre un correcte disseny.

Finalment, a partir de l'esquema lògic obtindrem el disseny físic de la base de dades, que a grans trets és l'adaptació de l'esquema lògic al SGBD^[4] concret (en aquest cas Oracle XE 11.2).

Un cop obtingut el disseny físic de la base de dades passarem a programar tots els procediments necessaris per a donar resposta a les necessitats de manipulació de les taules de la base de dades així com el correcte funcionament i actualització de les dades que conformen el magatzem. Així mateix, es generaran els procediments de consulta al magatzem de dades.

1.4 Planificació del Treball

El primer que hem de plantejar en el desenvolupament d'un treball com aquest, és els recursos necessaris per a dur-se a terme. En aquest aspecte diferenciar entre recursos de maquinari, programari i recursos temporals.

Pel que respecta al primer bloc, necessitarem una màquina on poder instal·lar tot el programari necessari per a dur a terme el projecte i sobre la qual poder córrer la base de dades que harem de desenvolupar. Atès que el volum de dades serà prou gran com per a poder efectuar totes les comprovacions necessàries sobre la base de dades, aquest no tindrà una magnitud desmesurada pel que qualsevol equip de gamma mitjana hauria de suportar perfectament a càrrega de treball que implica el desenvolupament del projecte.

[3] *Disseny lògic de bases de dades*: PID_00220510. Xavier Bargués Illa. Universitat Oberta de Catalunya.

[4] *Sistema Gestor de Base de Dades*: https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos

1.4.1 Programari necessari

Respecte al programari, hem de tenir en compte que aquest projecte requereix diversos programes de diferents finalitats. Passem doncs a examinar quines utilitats requerim per a desenvolupar tot el projecte i per quin programari ens decantarem:

- Planificador de tasques: El primer que necessitem fer al projecte és la planificació de les tasques del treball, de manera que necessitem un programari que pugui representar de manera gràfica i intuïtiva aquestes tasques al llarg del temps.

Microsoft Project 2016

(<https://products.office.com/es-es/project?tab=tabs-1>)

Aquest programari és propietat de Microsoft i s'executa amb una llicència d'estudiants lliurada per la UOC

- Editor de text: El projecte no només generarà la base de dades si no una memòria que recollirà tot el desenvolupament d'aquest. Per a dur-se a terme necessitem un editor de text.

Microsoft Word 2016

(<https://products.office.com/en-us/word>)

Aquest programari és propietat de Microsoft i s'executa amb una llicència personal.

- Editor de diagrames UML: Per a la fase conceptual de la base de dades, haurem de generar un diagrama conceptual del problema que estem tractant on es representin les entitats que en formen part, els seus atributs i les relacions entre elles.

Gliffy

(<https://www.gliffy.com/>)

Aquest programari és una eina en línia i s'executa amb una llicència de demostració de 15 dies lliurada després d'efectuar l'alta d'un nou usuari.

- Base de dades: Com a element central necessitem una base de dades al nostre equip per a poder generar l'estructura necessària i introduir el volum de dades requerit per a generar i testejar les consultes que se sol·liciten.

Oracle Database 11.2 Express Edition

(<http://www.oracle.com/technetwork/database/database-technologies/express-edition/downloads/index.html>)

Aquest programari és propietat de Oracle i gratuït.

- Sistema gestor de base de dades: Per a poder manipular la base de dades anteriorment esmentada necessitem un sistema gestor de base de dades.

Oracle SQLDeveloper

(<http://www.oracle.com/technetwork/es/developer-tools/sql-developer/downloads/index.html>)

Aquest programari és propietat de Oracle i és gratuït. Per a poder executar-se necessitarà la màquina virtual de Java.

- Maquina Virtual: Necessària per a poder executar el SGBD triat prèviament.

Java SE Development Kit 8

(<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>)

Aquest programari és propietat de Oracle i és gratuït.

- Editor per a la presentació: La tasca final projecte equival a generar una presentació que sintetitzi les memòries i sigui representativa de totes les tasques efectuades durant el desenvolupament de la base de dades.

Microsoft PowerPoint 2016

(<https://products.office.com/en-us/powerpoint>)

Aquest programari és propietat de Microsoft i s'executa amb una llicència personal.

- Captura de pantalla i edició de vídeo: Finalment necessitarem capturar certs elements del monitor del nostre equip i editar el vídeo per a poder muntar una presentació, de manera que farem ús d'un programa de captura de monitor i edició de vídeo.

Camtasia 9

(<https://www.techsmith.com/video-editor.html>)

Aquest programari és propietat de TechSmith i s'executa amb una llicència de trial de 30 dies.

1.4.2 Descomposició en tasques i planificació temporal

Ara que ja tenim clars els recursos que necessitarem per al desenvolupament del projecte en referència al maquinari i programari, dividirem el projecte en tasques.

Un cop desglossades totes les tasques que componen la realització del projecte, cal ubicar-les al llarg del temps en una correcta planificació temporal que ens permeti enllestir el projecte per la data sol·licitada amb un cert marge de maniobra així com assolir fites pel camí (en aquest cas, les marcarem en funció de les entregues parcials que hi ha en forma de PAC).

Per a fer aquest repartiment de forma òptima hem de tenir en compte la complexitat de cada tasca i el temps que preveiem dedicar-hi. A més hem de tenir en compte que existeixen dependències entre tasques, de manera que n'hi ha que no es poden iniciar fins a la finalització d'alguna altra, i de la mateixa manera existeixen tasques que es poden realitzar en paral·lel entre elles atès que malgrat han estat identificades com tasques diferenciades, tenen factors en comú i/o codependents.

- Feina inicial

Ubicació en el temps: 21/02/18 – 05/03/18

Cost en hores: 16h – 20h

- **Lectura de l'enunciat i documentació (4-6h):** El primer pas serà efectuar la lectura de l'enunciat del TFG per tal poder desgranar tota la informació que aquest conté i discernir dels problemes que planteja l'enunciat. També s'haurà de llegir documentació addicional com les pautes d'entrega del treball, les rúbriques de correcció, etc.

- **Pla de treball (12-14h):** Un cop llegida i assimilada tota la documentació inicial, serà necessari efectuar el pla de treball, desgranant la feina a efectuar en tasques i planificant la seva execució al llarg del semestre.

- Preparació de l'entorn

Ubicació en el temps: 06/03/18 – 11/03/18

Cost en hores: 4h – 8h

- **Tria del programari necessari (2-4h):** valorar entre el programari disponible a utilitzar per a poder dur a terme el treball.

- **Instal·lació i configuració (2-4h):** Caldrà, un cop triat el programari adient, instal·lar i configurar l'entorn i el programari per a poder dur a terme el treball.

- Disseny conceptual de la base de dades

Ubicació en el temps: 12/03/18 – 30/03/18

Cost en hores: 34h – 40h

- **Tria del conjunt de dades necessàries (10h):** El primer que haurem de tenir en compte és quines dades ha de desar la nostra base de dades per a poder generar la informació que la multinacional espera poder generar d'aquesta. És per això que haurem d'identificar cada entitat, atribut i relació de l'enunciat aportat.
- **Generació de diagrama UML (20-24h):** Un cop clares totes les entitats, atributs, relacions i restriccions de l'univers que compon el nostre problema, haurem de dissenyar la base de dades que serà capaç d'emmagatzemar i gestionar totes aquestes dades. Aquest disseny serà un mapa conceptual en UML.
- **Restriccions textuais (4h-6h):** Per finalitzar, caldrà tenir en compte totes aquelles restriccions que no puguin ser expressades gràficament al diagrama i efectuar-ne un llistat per a poder implementar-les quan generem la BD físicament.

- Disseny físic de la base de dades

Ubicació en el temps: 31/03/18 – 23/04/18

Cost en hores: 46h – 54h

- **Creació de taules (10-12h):** Un cop efectuat el disseny conceptual de la base de dades, hi haurà una transformació a sentències SQL que generin les entitats necessàries de manera que la realitat quedi representada de la mateixa manera que al diagrama conceptual.
- **Generació del volum de dades (4-6h):** Un cop estigui l'estructura de la base de dades creada, haurem de poblar les seves taules amb entrades per a poder generar consultes que ens retornin la informació que volem.
- **Creació dels procediments (28-30h):** El següent pas serà crear els procediments requerits per l'enunciat que retornin les dades que s'espera d'ells.
- **Afinació (4-6h):** Finalment es generaran tots aquells elements que siguin necessaris per a afinar la base de dades, com per exemple índexs addicionals, disparadors...

- Magatzem de dades

Ubicació en el temps: 24/04/18 – 02/05/18

Cost en hores: 22h – 28h

- **Estudi de les dades necessàries i estructura de les taules (10h):**
El primer que haurem de fer es estudiar quines dades es vol obtenir del magatzem en temps constant, assegurar-nos que aquestes dades estan presents a la nostra base de dades i determinar quines estructures tindran les taules que desaran aquestes dades per a poder donar resposta a les consultes especificades a l'enunciat en temps constant.
- **Creació de les taules (2-4h):** Un cop determinada la estructura de les taules, generarem les sentències SQL necessàries per a crear-les.
- **Creació dels procediments d'actualització (10h):** Caldrà crear els procediments necessaris per a inserir les dades a les taules que conformen el magatzem i determinar en quines "accions" s'han d'inserir aquestes dades. Aquestes "accions" es corresponen a altes, baixes i/o modificacions d'entrades a taules de la base de dades.
- **Generació de les consultes (4h):** Creació de les consultes que retornaran allò que l'enunciat del treball requereix.

- Prova de la base de dades

Ubicació en el temps: 03/05/18 – 07/05/18

Cost en hores: 10h – 12h

- **Creació de jocs de proves (6h):** El primer que farem per a testejar la base de dades serà crear jocs de proves per a posar la BD en casos límits, extrems i/o poc comuns per assegurar-nos que aquesta sempre retorna la informació que s'espera d'acord amb la consulta efectuada o al procediment executat. També es faran jocs de proves convencionals perquè es puguin testejar les seves funcionalitats de manera còmoda.
- **Prova de la base de dades (4-6h):** Un cop amb els jocs de proves generats, arribarà el moment de consultar a la base de dades i veure que els resultats que genera són els esperats.

- Documentació a lliurar

Ubicació en el temps: 21/02/18 – 11/06/18

Cost en hores: 158h – 164h

- **Pla de treball (12-14h):** Document on es desgranaran totes les tasques a efectuar durant el projecte i es planificaran al llarg del temps del què es disposa.
- **Memòria (120h):** Aquesta s'anirà efectuant al llarg de tot el desenvolupament del TFG, i contindrà tots els procediments seguits, dubtes sorgits, solucions, decisions, alternatives i en resum un seguiment del desenvolupament de totes les fases del treball.
- **Base de dades (18-20h):** La base de dades és lliura en forma de scripts de creació de taules, d'inserció del volum de dades, de creació dels procediments...
- **Jocs de proves (8-10h):** Es lliurarà també els jocs de proves necessaris per a provar les funcionalitats i el correcte funcionament de la base de dades i del magatzem de dades.

- Presentació en vídeo

Ubicació en el temps: 08/05/18 – 28/05/18

Cost en hores: 30h – 36h

- **Síntesis del TFG i generació de diapositives (8-10h):** El primer pas serà efectuar un resum del treball en diapositives de suport que es mostraran al vídeo de defensa del treball.
- **Creació del guió (4-6h):** S'ha de generar un guió d'allò que es pensa explicar al vídeo de defensa per tal que aquest quedi ordenat, sintetitzat i coherent.
- **Preparació de l'entorn (4h):** Caldrà preparar l'entorn per a la gravació, tant en l'àmbit de programari (triad un programa d'edició de vídeo i/o de captura) com de maquinari (càmera, micròfon...)
- **Preparació de la presentació (4h):** Temps que s'utilitzarà per a estudiar el guió i preparar la gravació del vídeo de defensa.
- **Gravació i muntatge (10-12h):** Gravació del vídeo

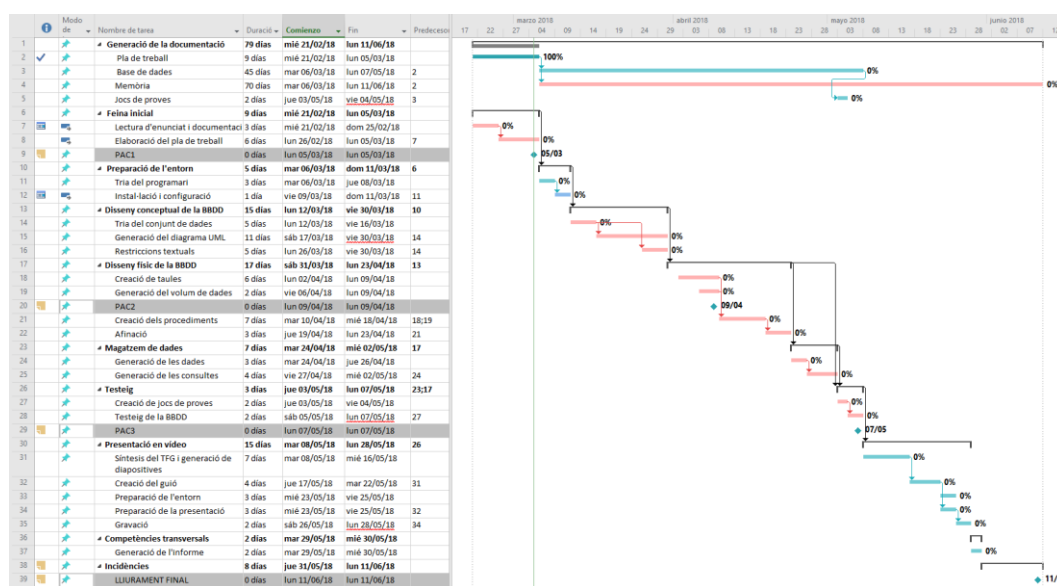
- Avaluació de les competències transversals

Ubicació en el temps: 29/02/18 – 30/06/18

Cost en hores: 2h – 4h

- **Generació de l'informe (2-4h):** Cal generar un informe d'autoavaluació d'aquelles competències transversals emprades al llarg del grau i posades de manifest en l'execució del TFG.

Així doncs la planificació temporal que s'ha fet a l'inici del projecte queda de la següent manera:



Com és obvi, al llarg del semestre intentarem seguir de la manera més acurada aquesta planificació temporal per a poder arribar als lliuraments amb les fites que ens hem marcat.

No obstant això, per desconeixement de les tasques a realitzar i imprevistos el desenvolupament final pot variar respecte a aquesta planificació. Totes aquestes variacions de temps i tasques seran exposades a l'apartat Conclusions d'aquest treball, fent una comparativa entre el temps planificat i el que realment ha sigut necessari per a cada tasca.

1.5 Breu sumari de productes obtinguts

Com a producte principal obtingut després del desenvolupament d'aquest projecte, obtindrem una base de dades relacional amb les estructures i funcionalitats necessàries per a cobrir les necessitats que l'empresa requereix. Això és totes les taules i tots els procediments necessaris per a poder gestionar les dades que aquesta ha de desar.

Per una altra banda, la base de dades comptarà amb un magatzem de dades que donarà resposta a les consultes estipulades en un temps constant. Per aquesta vessant, obtindrem les estructures necessàries per a poder donar aquesta informació en el temps requerit així com tots els procediments necessaris per a un correcte funcionament. En aquest cas els procediments seran tant els de gestió de les dades com els de consulta.

Finalment la base de dades comptarà amb un volum inicial de dades i una bateria de jocs de proves per a cadascuna de les funcionalitats que aquest ofereix.

En el seu conjunt, tot el que acabem d'exposar conformaria la base de dades com a producte obtingut en forma d'arxius amb comandes SQL preparats per a ser executats al SGBD seleccionat.

Finalment i acompanyat a aquests arxius de creació i prova de la base de dades, es generarà aquesta memòria on s'explicarà cada pas seguit pel desenvolupament del projecte, cada decisió presa i l'execució i resultats dels jocs de proves. També es generarà una presentació en vídeo que sintetitzi el contingut de la memòria a tall de defensa del projecte.

1.6 Breu descripció dels altres capítols de la memòria

Els capítols que veurem a continuació estan dividits segons les fases de desenvolupament d'una base de dades i, posteriorment, uns capítols que mostren les proves a la base de dades i les conclusions obtingudes amb el desenvolupament del projecte. Posem de manifest la figura central dels capítols de disseny conceptual, lògic i físic de la base de dades i la creació dels procediments i del magatzem atès que aquest són el cor del projecte, els capítols que esdevenen en el producte que es genera amb l'execució del TFG.

A grans trets, el contingut dels capítols d'aquesta memòria és el següent:

- Disseny Conceptual: Diagrama UML conceptual de la nostra base de dades. En aquest capítol es posaran de manifest les entitats que formen part de la realitat del problema, els seus atributs i les relacions que les uneixen. També s'exposaran i es justificaran aquelles decisions de disseny que han aparegut en la creació del mapa.
- Disseny Lògic: Traducció del diagrama a unes sentències que es troben a mig camí entre el mapa conceptual i les sentències SQL que crearan la base de dades. Aquest capítol també inclou una anàlisi de la correctesa de la base de dades fent ús de la teoria de la normalització.
- Disseny Físic: Sentències SQL que crearan les estructures de la nostra base de dades. És la traducció del model lògic anterior a un llenguatge específic pel sistema gestor de base de dades.
- Procediments: Creació de tots els procediments necessaris per a gestionar les dades que es desaran a les taules creades. Aquests procediments poden ser d'alta, de baixa o de modificació d'alguns atributs.
- Data warehouse: Creació tant de les estructures que desaran la informació que ha de ser lliurada en temps constant, com de tots aquells procediments necessaris per a mantenir aquestes estructures correctament actualitzades en funció de les accions que es generin sobre la base de dades operativa. En aquest cas, es tracta de procediments que s'encarregaran de modificar aquestes taules del magatzem com de procediments de consulta d'aquestes.
- Proves: Planificació i creació d'una bateria de proves que s'encarregarà de demostrar el correcte funcionament de la base de dades per tots els casos i funcionalitats que aquesta té.
- Conclusions: Finalment, unes conclusions sobre el desenvolupament del projecte així com una anàlisi crítica de l'assoliment d'objectius i el compliment de la planificació temporal.

2. Disseny Conceptual

2.1 Identificació d'entitats, atributs i relacions

El primer que hem de fer per a poder plasmar la realitat del problema en un mapa conceptual que ens servirà com a base per a poder generar la nostra base de dades, és determinar els elements que compondran aquest diagrama.

En el cas que ens ocupa, hem de generar un diagrama UML^[5] per a una base de dades ER, de manera que en aquest diagrama hauran d'aparèixer entitats que constaran d'un llistat d'atributs i les relacions que existeixen entre aquestes.

Un cop esclarit que és el que busquem, em de determinar quins elements de la realitat poden ser considerats entitats, és a dir elements que siguin clarament diferenciables entre ells. A continuació haurem de determinar els atributs d'aquestes entitats, és a dir, aquells elements que les defineixen i quins d'aquests atributs les diferencien entre si (seran candidats per a ser la clau primària de la classe, que serà representada al diagrama amb <P>).

Finalment un cop trobades totes les entitats i els seus atributs, caldrà discernir quines relacions rellevants per al problema que la base de dades ha de resoldre existeixen entre les diferents entitats.

Passem doncs a elaborar un llistat d'entitats i relacions. Cal dir que aquest llistat és una primera aproximació del que serà la base de dades, però no el definitiu perquè aquest poden sofrir canvis fent desaparèixer entitats, convertint atributs en entitats pròpies o variant les relacions que en primera instància determinem per motius de disseny, traducció o funcionalitat final de la base de dades. Tots aquests conceptes seran explicats en punts posteriors del treball.

- **Usuaris del sistema:** correu corporatiu que els identifica, data del registre, data de baixa, seu a la qual pertanyen.
- **Seus corporatives:** codi identificatiu, direcció de la seu, tipologia corporativa que presenta, país on està ubicada.
- **Països:** nom del país
- **Interaccions dels usuaris:** data de la interacció, usuari que l'ha generat, acció generada.
- **Catàleg de Serveis:** codi identificatiu del servei, descripció, temps mig necessari per a la seva aplicació, grup al qual pertany el servei, prioritat, grups de suport que ofereixen el servei.

[5] Llenguatge UML: https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado

- **Grup de suport:** codi del grup, nivell del grup, tècnics que el componen, tiquets associats, serveis que ofereixen, assignacions amb tiquets.
- **Tiquets:** número identificatiu, data de la creació, tipus, canal d'entrada, mètode de creació, prioritat, servei sol·licitat, missatges que produeix, canvis d'estat que sofreix, assignacions amb els grups de suport,
- **Canvis d'estat:** tècnic que el genera, data del canvi, tiquet que sofreix en canvi d'estat, estat actual.
- **Missatges:** codi identificatiu, data del missatge, tiquet al qual pertany, emissor del missatge.

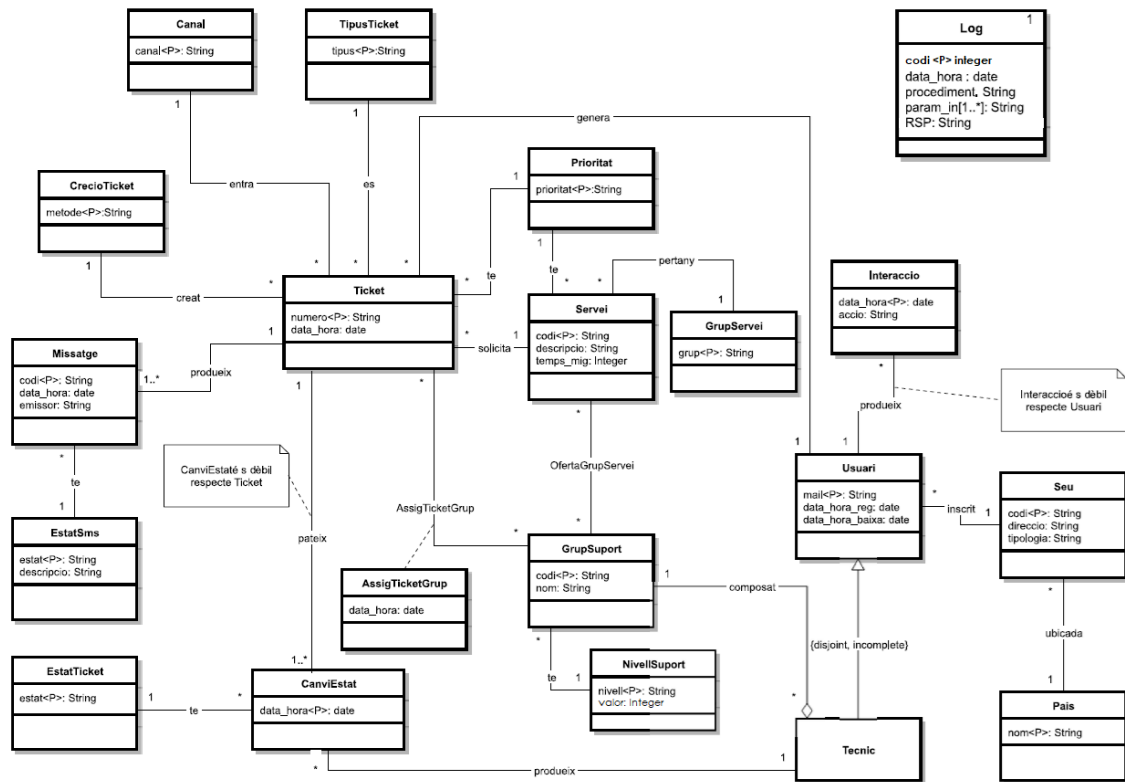
Pel que respecta a les relacions, dependrà molt que acabem determinant com entitat o atribut en el diagrama, però algunes d'aquestes relacions són molt clares i altres han sortit en la mateixa descripció que acabem de fer de les entitats, tals com:

- Les seus estan **ubicades** a països.
- Els usuaris estan **inscrits** a les seus.
- Els usuaris **produeixen** interaccions.
- Els usuaris tècnics **generen** canvis d'estat a tiquets.
- Un grup de suport està **compost** per usuaris tècnics.
- Un tiquet **sol·licita** un determinat servei.
- Un grup de suport **oferta** uns determinats tipus de serveis.
- Els usuaris **generen** tiquets.
- Un tiquet **produeix** un seguit de missatges.
- Un grup de suport està **assignat** a un seguit de tiquets.
- Un tiquet està **assignat** a un seguit de grups de suport.

Amb tot això en ment i respectant el fet que aquestes entitats, atributs i relacions són un esbós previ a la construcció del mapa que pot variar pels factors ja exposats, passem a presentar el disseny resultant del mapa conceptual, tot exposant les decisions de disseny preses, les versions prèvies del mapa i les restriccions no presents gràficament sobre el mateix.

2.2 Diagrama conceptual

A continuació podem observar com resulta el mapa conceptual que modela el nostre problema utilitzant el llenguatge UML.



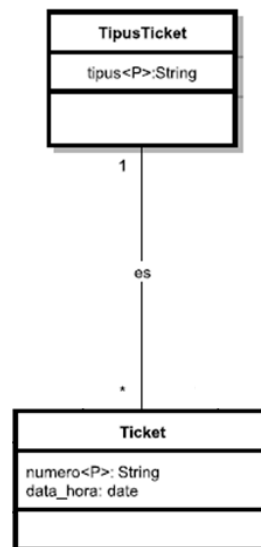
**Aquest mapa és la segona versió que es va generar en el seu desenvolupament. S'adjunta la primera versió del mateix a l'[Annex](#). Tots els canvis soferts entre versions seran explicats al següent punt del treball: [Decisions de disseny](#).*

2.3 Decisions de disseny

Un cop presentat el diagrama conceptual que representa la primera fase de la nostra base de dades, passem a reflexionar i explicar algunes de les decisions de disseny preses en el diagrama i sospesar les alternatives possibles, així com els fets que han fet que ens decantéssim pel disseny definitiu.

Si bé es cert que tot és susceptible a ser examinat, ens basarem en aquells detalls del disseny els quals en el moment del seu plantejament es van contemplar diferents alternatives.

2.3.1 Tipus de tiquet



El primer dubte que va sorgir durant el desenvolupament del diagrama és si era necessari tractar els tipus de tiquet com a entitats diferenciades. És a dir, si valia la pena que **'Ticket'** fos una generalització de les classes **'Petició'** i **'Incidència'** fent que aquestes fossin filles de tiquet.

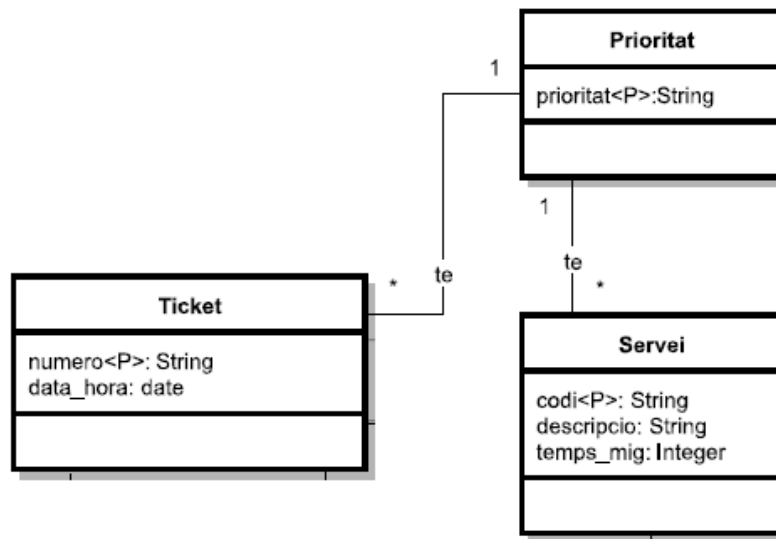
Aquesta idea va ser descartada atès que en el problema que ens ocupa, el concepte de les entitats pròpies d'incidència i petició no ens importen, ja que el problema que tenim entre mans és el d'encarregar-nos dels tiquets que generen els usuaris. Aquesta idea va ser traslladada i aprovada pel consultor del treball.

Un cop determinat que no era necessari contemplar les incidències i les peticions com a entitats pròpies, es va determinar que aquest havia de ser un atribut de la classe **'Ticket'**. En aquest punt, es important assegurar-nos que els tipus seran introduïts sempre de manera correcta pels usuaris, de manera que és important que els possibles valors dels atributs estiguin presents a la base de dades. Sent així, podem considerar el tipus com un atribut i fer una enumeració dels possibles valors que aquest pot prendre o bé considerar el tipus com una entitat relacionada amb el tiquet.

En aquest cas tal com es pot observar al diagrama el tipus del tiquet és una entitat amb un atribut que correspon al nom del tipus. Aquesta decisió de disseny es va prendre pel fet que d'aquesta manera el tipus es traduirà com una taula a la base de dades, on en un futur serà molt més fàcil introduir nous tipus de tiquets.

Cal veure que en cas que aquest s'hagués traduït com un atribut de **'Ticket'** amb una enumeració dels possibles valors, aquesta enumeració s'hauria de traduir com una funció o comprovació a la traducció de la taula que correspon a l'entitat **'Ticket'**, fent que introduir al sistema un nou tipus de tiquet sigui molt més complex que el simple fet d'introduir un nou valor a la taula **'TipusTicket'**.

2.3.2 Prioritats



De manera molt similar a com ja hem vist en el punt anterior, cal determinar com es representa la prioritat tant en el cas dels tiquets com en el cas dels serveis.

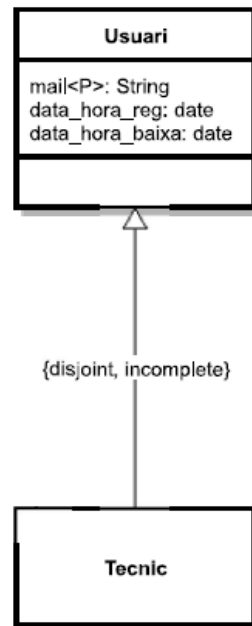
En aquest cas, és aplicable el que hem pogut veure respecte els tipus del tiquet pel que fa a la necessitat de tenir presents els possibles valors dins la base de dades per assegurar-nos que aquest s'introdueixen sempre de manera correcta.

En aquest sentit, la creació d'una entitat pròpia també es pot justificar amb la futura traducció d'aquest fet a la base de dades, ja que serà molt més fàcil tenir una entitat pròpia per si en un futur es vol crear un nou grau de prioritat, només serà necessari introduir un nou valor a la taula '**Prioritat**' i no haurèm de reprogramar una funció de comprovació o la mateixa definició de les taules com si passaria si traduïssim aquest fet com una enumeració de possibles valors d'un atribut.

Això no obstant, la decisió sobre el fet de si traslladar aquesta necessitat en forma d'entitat pròpia o d'enumeració de valors en aquest cas pren un valor afegit que no és altre que el fet que la prioritat està present en més d'una entitat del nostre disseny (concretament sobre l'entitat '**Ticket**' i '**Servei**'), i si bé és cert que aquestes no tenen per què coincidir entre elles, els valors que poden prendre en els dos casos són els mateixos.

Amb tot això s'ha decidit interpretar la prioritat com una entitat pròpia amb un sol atribut que representa els possibles valors que aquesta pot prendre al sistema.

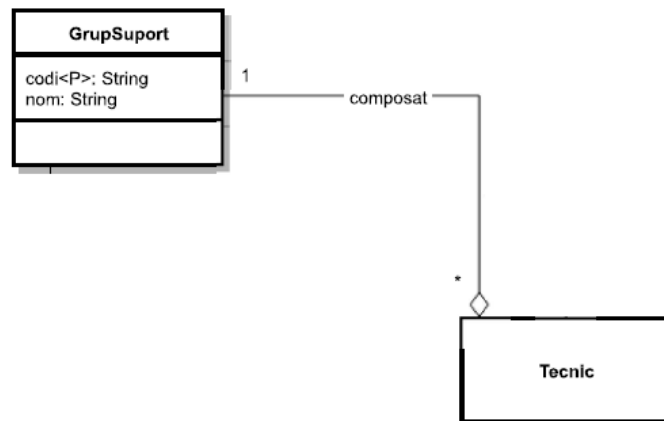
2.3.3 Usuaris tècnics



En aquest cas ens trobem amb dubte que sorgeix en tenir un tipus d'usuari específic.

A l'enunciat se'ns parla que els usuaris han d'estar registrats al sistema per a poder generar tiquets i rebre suport per part d'uns altres usuaris que anomenarem tècnics. Davant d'aquest fet, i tenint en compte que les dades més genèriques dels usuaris no cal desar-les perquè aquestes es troben a una base de dades global de l'empresa, observem que aquests usuaris tècnics no tenen cap atribut específic que els diferenciï de la resta d'usuaris del sistema.

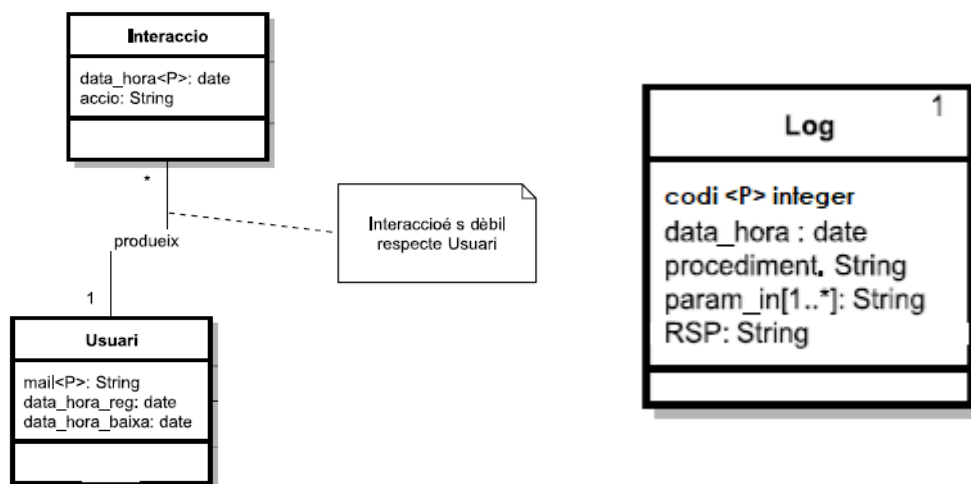
Això no obstant, el detonant que ens porta a prendre la decisió de considerar els tècnics com una especificació dels usuaris genèrics és que aquests tenen una relació específica i única amb els grups de suport tal com es pot veure a consignació.



Tal com podem observar, els grups de suport estan compostos per usuaris tècnics del sistema, per tant existeix una relació de composició entre les dues entitats. Aquesta relació només es pot donar amb els usuaris tècnics, ja que un usuari genèric no pot formar part d'un grup de suport. Amb això s'ha adoptat la decisió de considerar els usuaris tècnics com una subclasse dels usuaris corrents.

Com es pot apreciar al disseny, aquesta classe serà incompleta (atès que no tots els usuaris són tècnics) i disjunta (aquest valor de moment no té massa sentit existint només una subclasse de treballadors i en cas de voler introduir una altra subclasse en un futur inclús podia canviar a overlapping).

2.3.4 Registres



Segons el plantejament del problema, al sistema han d'existir uns "registres" de certes interaccions que es donaran amb la base de dades. Concretament han d'existir dos registres tal com podem observar a la imatge superior. Anem doncs a examinar-los per separat.

- **Registre d'interaccions dels usuaris del sistema:** En primer lloc ens trobem un registre que ha de desar totes les interaccions que els usuaris del sistema efectuïn. Aquest fet s'ha traduït amb una entitat 'Interacció' relacionada amb els usuaris, reflectint que aquests últims produeixen interaccions. Cal recalcar que una interacció s'identifica per l'usuari que l'ha causat i el moment del temps en què s'ha produït, de manera que tal com podem veure 'Interacció' és una entitat dèbil respecte l'entitat 'Usuari'.
- **Registre de crides a procediments:** Aquest segons registre és el registre que captura totes les crides als procediments de la base de dades. Aquest registre servirà per comprovar el correcte funcionament d'aquestes funcions cridades en algun moment del temps. Tal com es pot observar al disseny, es considera que només existirà una instància d'aquesta classe al sistema.

2.3.5 Enumeracions VS Entitats pròpies

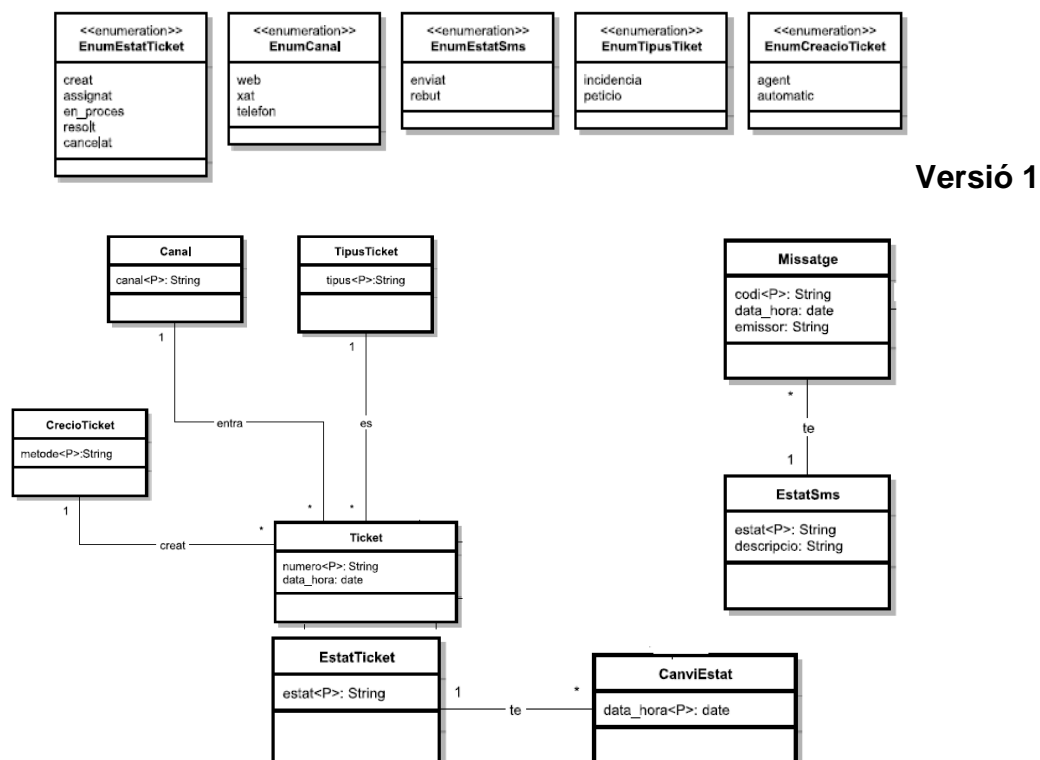
En aquest punt aglomerarem diversos casos en els quals la decisió a prendre era la mateixa per a tots ells. Aquests casos són aquelles en els que tenim un atribut que pot prendre un conjunt determinat de valors, i volem assegurar-nos que aquests valors estan presents a la base de dades per tal de cerciorar-nos que aquests sempre s'introduiran de manera correcta.

Davant aquest fet tal com ja hem vist en punts anteriors podem optar per fer una enumeració de valors que es traduirà com a una funció de comprovació en la inserció de registres a les taules que continguin l'atribut que pot tenir aquest conjunt de valors, o bé definir aquests valors com registres d'una entitat amb un atribut que contingui tantes entrades com possibles valors pugui prendre el concepte que representa.

En el primer cas, si volem afegir un nou valor a un conjunt, haurem de redefinir la funció de comprovació, podent afectar inclús a la definició de la mateixa taula de l'entitat.

En el segon (cas pel qual es va optar finalment) l'ampliació d'aquests valors és molt més senzilla perquè a efectes pràctics es resumeix a introduir un nou registre per a la taula que representa aquest conjunt de valors.

A continuació podem observar totes aquelles enumeracions que es van reconsiderar a entitats pels motius prèviament esmentats. La primera imatge correspon a la primera versió del mapa conceptual que es pot consultar a l'[Annex](#).



2.4 Restriccions textuais

Finalment només resta fer una anàlisi d'aquelles restriccions que existeixen al problema que estem tractant al món real però que no han pogut ser plasmades al diagrama conceptual. Aquestes restriccions hauran de ser tingudes en compte a l'hora de crear les taules de la base de dades o inclús a l'hora de generar els procediments per a tractar aquestes dades. En la seva majoria, finalment seran traduïdes com a comprovacions prèvies a l'alta, baixa i/o modificació de les dades. Així doncs el llistat de restriccions textuais és el següent:

- ***RT1: Donat un canvi d'assignació d'un grup de suport a un tiquet, el nivell de suport del nou grup de suport assignat ha de ser superior al nivell de suport del grup de suport assignat prèviament.***

Això és perquè el flux de resolució dels problemes especifica que un tiquet serà tractat per un grup de suport amb nivell de suport L1. Si aquest no té èxit en la seva resolució, el tiquet haurà de ser tractat per un grup de suport amb nivell de suport L2 i així successivament fins a PM.

Per tant s'ha de controlar que les assignacions de grups a un mateix tiquet segueixi aquesta relació entre els diferents nivells de suport: L1 → L2 → L3 → PM

- ***RT2: El tècnic que efectua un canvi d'estat sobre un tiquet en un moment concret, ha de pertànyer necessàriament al grup de suport que està assignat al tiquet en el moment del canvi d'estat.***

Aquesta restricció expressa que només els tècnics que formin part del grup de suport que s'encarrega del tiquet en aquell moment, poden efectuar canvis d'estat sobre el tiquet en qüestió.

- ***RT3: L'emissor d'un missatge pot ser o bé l'usuari que ha generat el tiquet associat a aquest missatge o bé el sistema.***

En aquesta restricció s'especifica que l'emissor d'un missatge per a un tiquet determinat no podrà ser qualsevol usuari de la base de dades sinó que ha de ser l'usuari que ha generat el tiquet que té associat aquest missatge, o bé una entitat genèrica 'sys' associada al sistema, atès que els missatges generen una comunicació entre l'usuari i el sistema.

- ***RT4: La data/hora de registre d'un usuari al sistema ha de ser necessàriament menor a la data/hora de baixa pel mateix usuari en cas que aquesta última no sigui nul·la.***

Òbviament en cas que un usuari tingui una data/hora de baixa del sistema, aquesta data haurà de ser necessàriament superior a la data/hora en què es va donar l'usuari d'alta, atès que no tindria sentit que l'interval de temps en què l'usuari ha estat present al sistema fos negatiu.

- ***RT5: El grup de suport assignat a un tiquet determinat ha d'oferir el servei que aquest tiquet sol·licita.***

Atès que no tots els grups de suport ofereixen tots els serveis disponibles al sistema, hem d'assegurar-nos que no es doni el cas que un grup de suport sigui assignat a un tiquet que sol·licita un servei que aquest grup no pot oferir.

3. Disseny Lògic

3.2 Traducció del model conceptual al model lògic

En aquest punt el que farem serà la traducció del model conceptual en UML presentat al punt anterior al model lògic. Aquest model dependrà del tipus de base de dades que farem servir (en aquest cas una base de dades relacional) però no tindrà en compte les especificacions concretes de la implementació del SGBD.

Així doncs passem a veure el resultat de la traducció. Abans només recordar que tot i que fem servir la lletra subratllada per indicar que aquell atribut o conjunt d'atributs són la clau primària i la lletra en **negreta** per indicar que aquell atribut no pot prendre valor NULL, ho indicarem també com a restriccions sota cada relació:

Pais (nom)
{nom} is PRIMARY KEY

Seu (codi, **direccio**, tipologia, **pais**)
{codi} is PRIMARY KEY
{pais} is FOREIGN KEY to pais {nom}
{direccio} is NOT NULL
{pais} is NOT NULL

Usuari (mail, **data_hora_reg**, data_hora_baixa, **seu**)
{mail} is PRIMARY KEY
{seu} is FOREIGN KEY to seu {codi}
{data_hora_reg} is NOT NULL
{seu} is NOT NULL
RT4: si ({data_hora_baixa} != NULL) llavors {data_hora_baixa} > {data_hora_reg}

Interaccio (usuari, **data_hora**, **accio**)
{usuari, data_hora} is PRIMARY KEY
{usuari} is FOREIGN KEY to Usuari {mail}
{data_hora} is NOT NULL
{accio} is NOT NULL

NivellSuport (nivell, **valor**)
{nivell} is PRIMARY KEY
{valor} is NOT NULL

GrupSuport (codi, **nom**, **nivell**)
{codi} is PRIMARY KEY
{nivell} is FOREIGN KEY to NivellSuport {nivell}
{nom} is NOT NULL
{nivell} is NOT NULL

Tecnic (mail, **grup**)

{mail} is PRIMARY KEY

{mail} is FOREIGN KEY to Usuari {mail}

{grup} is FOREIGN KEY to GrupSuport {codi}

{grup} is NOT NULL

Prioritat (prioritat)

{prioritat} is PRIMARY KEY

GrupServei (grup)

{grup} is PRIMARY KEY

Servei (codi, descripcio, **grup**, **temps_mig**, **prioritat**)

{codi} is PRIMARY KEY

{grup} is FOREIGN KEY to GrupServei {grup}

{prioritat} is FOREIGN KEY to Prioritat {prioritat}

{grup} is NOT NULL

{temps_mig} is NOT NULL

{prioritat} is NOT NULL

OferetaGrupServei (grup, servei)

{grup, servei} is PRIMARY KEY

{grup} is FOREIGN KEY to GrupSuport{codi}

{servei} is FOREIGN KEY to Servei{codi}

TipusTicket (tipus)

{tipus} is PRIMARY KEY

Canal (canal)

{canal} is PRIMARY KEY

CreacioTicket (metode)

{metode} is PRIMARY KEY

Log (codi, **data_hora**, **procediment**, **param_in**, **RSP**)

{codi} is PRIMARY KEY

{data_hora} is NOT NULL

{procediment} is NOT NULL

{param_in} is NOT NULL

{RSP } is NOT NULL

Ticket (numero, **data_hora_creacio**, **servei**, **prioritat**, **tipus**, **canal**, **creacio**, **usuari**)

{numero} is PRIMARY KEY
{servei} is FOREIGN KEY to Servei {codi}
{prioritat} is FOREIGN KEY to Prioritat {prioritat}
{tipus} is FOREIGN KEY to TipusTicket {tipus}
{canal} is FOREIGN KEY to Canal {canal}
{creacio} is FOREIGN KEY to CreacioTicket {creacio}
{usuari} is FOREIGN KEY to Usuari {mail}
{data_hora_creacio} is NOT NULL
{servei} is NOT NULL
{prioritat} is NOT NULL
{tipus} is NOT NULL
{canal} is NOT NULL
{creacio} is NOT NULL
{usuari} is NOT NULL

AssigTicketGrup (ticket, grup, **data_hora**)

{ticket, grup} is PRIMARY KEY
{ticket} is FOREIGN KEY to Ticket {numero}
{grup} is FOREIGN KEY to GrupSuport {codi}
RT1: {grup.nivell} actual ha de ser superior a {grup.nivell} anterior si existeix un anterior
RT5: {ticket→Ticket→Servei.codi} ha d'estar a {grup→GrupSuport→Servei.codi}

EstatTicket (estat, **descripcio**)

{estat} is PRIMARY KEY

CanviEstat (ticket, data_hora, **tècnic**, **estat**)

{ticket, data_hora} is PRIMARY KEY
{ticket} is FOREIGN KEY to Ticket {numero}
{tecnic} is FOREIGN KEY to Tecnic {codi}
{estat} is FOREIGN KEY to EstatTicket {estat}
{data_hora} is NOT NULL
{tecnic} is NOT NULL
{estat} is NOT NULL
RT2: {tecnic→GrupSuport.codi} == {ticket→AssigTicketGrup.grup} per la data actual

EstatSms (estat, **descripcio**)

{estat} is PRIMARY KEY

Missatge (codi, **data_hora**, **emissor**, **estat**, **ticket**)

{codi} is PRIMARY KEY
{estat} is FOREIGN KEY to EstatSms {estat}
{ticket} is FOREIGN KEY to Ticket {numero}
{data_hora} is NOT NULL
{emissor} is NOT NULL
{estat} is NOT NULL
{ticket} is NOT NULL
RT3: ({emissor} == 'SYS' || {emissor} == {ticket→Ticket.usuari})

Tal com podem observar, hem definit les relacions que formaran part de la nostra base de dades a partir del diagrama conceptual que hem efectuat en la primera part del treball. Aquestes relacions, formades per un seguit d'atributs i relacions (en el sentit més literal de la paraula) transformades en claus foranes, consten d'atributs que les identifiquen definits com a claus primàries, i atributs que no poden prendre valors nuls.

Un cop obtingut el model lògic, i previ a la traducció d'aquestes relacions a sentències SQL, cal observar si existeix alguna anomalia originada en les dependències existents entre els diferents atributs. Per a fer-ho utilitzarem la teoria de la normalització.

3.3 Normalització

El primer que cal recordar és que la teoria de la normalització defineix una sèrie de nivells, anomenats formes normals, que eliminen progressivament determinades dependències que són causants de diferents anomalies. Aquestes formes normals són inclusives; és a dir, si una relació compleix les condicions d'un determinat nivell, també compleix les condicions de tots els nivells anteriors^[6]. Aquests nivells són els següents:

- **Primera forma normal**^[7]: Cap atribut de la relació és ell mateix una relació ni descomponible ni amb multiplicitat de valors.
- **Segona forma normal**^[8]: Primera forma normal + Tot atribut que no forma part d'una clau candidata depèn completament de totes les claus candidates de la relació.
- **Tercera forma normal**^[9]: Segona Forma Normal + Cap atribut que no forma part d'una clau candidata depèn d'un conjunt d'atributs que en conté algun que no forma part d'una clau candidata.
- **Forma normal de Boyce-Codd**^[10]: Tercera Forma Normal + Els determinants de totes les dependències que presenta la relació en són claus candidates.

Per tant el que es farà a continuació serà, en primer lloc, examinar les dependències entre els atributs de les relacions i posteriorment s'analitzaran aquestes dependències per establir si la relació es troba en FNBC. Aquest procés es farà per a cadascuna de les relacions resultants de la traducció al model lògic.

[6] Disseny lògic de bases de dades: PID_00220510. Xavier Bargués Illa. Universitat Oberta de Catalunya.

[7] Primera forma normal: https://es.wikipedia.org/wiki/Primera_forma_normal

[8] Segona forma normal: https://es.wikipedia.org/wiki/Segunda_forma_normal

[9] Tercera forma normal: https://es.wikipedia.org/wiki/Tercera_forma_normal

[10] Forma normal de Boyce-Codd: https://es.wikipedia.org/wiki/Forma_normal_de_Boyce-Codd

3.3.1 Pais

- Relació: Pais (nom)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.2 Seu

- Relació: Seu (codi, direccio, tipologia, pais)
- Dependències: {codi} → {direcció}
 {codi} → {tipologia}
 {codi} → {pais}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.3 Usuari

- Relació: Usuari (mail, data_hora_reg, data_hora_baixa, seu)
- Dependències: {mail} → {data_hora_reg}
 {mail} → {data_hora_baixa}
 {mail} → {seu}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.4 Interaccio

- Relació: Interaccio (usuari, data_hora, accio)
- Dependències: {usuari, data_hora} → {accio}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn dels atributs que sí ho són. Per tant es troba en FNBC.

3.3.5 NivellSuport

- Relació: NivellSuport (nivell, valor)
- Dependències: {nivell → valor}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.6 Grup Suport

- Relació: GrupSuport (codi, nom, nivell)
- Dependències: {codi → nom}
 {codi → nivell}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.7 Tecnic

- Relació: Tecnic (mail, grup)
- Dependències: {mail} → {grup}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.8 Prioritat

- Relació: Prioritat (prioritat)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.9 GrupServei

- Relació: GrupServei (grup)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.10 Servei

- Relació: Servei (codi, descripcio, grup, temps_mig, prioritat)
- Dependències: {codi} → {descripcio}
 {codi} → {grup}
 {codi} → {temps_mig}
 {codi} → {prioritat}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.11 OferetaGrupServei

- Relació: OferetaGrupServei (grup, servei)
- Dependències: Inexistents
- Anàlisi: La clau primària està composta dels dos atributs que té la relació, per tant es troba en FNBC.

3.3.12 TipusTicket

- Relació: TipusTicket (tipus)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.13 Canal

- Relació: Canal (canal)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.14 CreacioTicket

- Relació: CreacioTicket (metode)
- Dependències: Inexistents
- Anàlisi: Al ser una entitat amb un sol atribut, no existeixen dependències i per tant està en FNBC.

3.3.15 Log

- Relació: Log (data_hora, procediment, param_in, RSP)
- Dependències: {codi} → {data_hora}
 {codi} → {procediment}
 {codi} → {param_in}
 {codi} → {RSP}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.16 Ticket

- Relació: Ticket (numero, data_hora_creacio, servei, prioritat, tipus, canal, creacio, usuari)
- Dependències: {numero} → {data_hora_creacio}
 {numero} → {servei}
 {numero} → {prioritat}
 {numero} → {tipus}
 {numero} → {canal}
 {numero} → {creacio}
 {numero} → {usuari}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.17 AssigTicketGrup

- Relació: AssigTicketGrup (ticket, grup, data_hora)
- Dependències: {ticket, grup} → {data_hora}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn dels atributs que sí ho són. Per tant es troba en FNBC.

3.3.18 EstatTicket

- Relació: EstatTicket (estat, descripcio)
- Dependències: {estat} → {descripcio}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.19 CanviEstat

- Relació: CanviEstat (ticket, data_hora, tècnic, estat)
- Dependències: {ticket, data_hora} → {tecnic}
{ticket, data_hora} → {estat}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.20 EstatSms

- Relació: EstatSms (estat, descripcio)
- Dependències: {estat} → {descripcio}
- Anàlisi: Només existeix una dependència on l'atribut que no és clau candidata depèn de l'atribut que si ho és. Per tant es troba en FNBC.

3.3.21 Missatge

- Relació: Missatge (codi, data_hora, emissor, estat, ticket)
- Dependències: {codi} → {data_hora}
{codi} → {emissor}
{codi} → {estat}
{codi} → {ticket}
- Anàlisi: Només existeixen dependències on els atributs que no són claus candidates depenen de l'atribut que si ho és. Per tant es troba en FNBC.

Arribats a aquest punt, podem comprovar com totes les relacions del nostre model lògic de la base de dades es troben en forma normal de Boyce-Codd, i per tant ja podem passar a traduir aquest model a sentències SQL per a crear les taules que representaran cada entitat de les que acabem de presentar.

4. Disseny Físic

Ara ens trobem en la tercera etapa del desenvolupament de la nostra base de dades. Fins ara hem plasmat la realitat en un mapa conceptual representant les entitats, els seus atributs i les relacions existents entre elles. Posteriorment hem traduït aquest mapa conceptual en un seguit de entitats amb unes restriccions concretes a mode de model lògic de la base de dades, que recordem que si té en compte el tipus de base de dades que farem servir, no té en compte el SGBD que gestionarà la base de dades.

4.1 Traducció del model lògic a sentències SQL

En aquest apartat presentarem la traducció del model lògic al model físic de les sentències necessàries per a la creació de la base de dades. Atès que el resultat és el codi SQL i per tal de no saturar la memòria amb codi, aquesta traducció s'ubicarà a l'[Annex](#).

Observant la traducció, podem veure la definició del tipus de dades de cada atribut i les restriccions que aquests tenen. Aquestes restriccions són de tipus NOT NULL, definicions de claus primàries, definicions de claus foranes i algun control. Cal aclarir que la sintaxi per a declarar aquestes restriccions pot ser més directa i curta, per exemple si decidim no donar noms a les CONSTRAINTS. Amb tot i amb això s'ha optat per fer-ho així per seguir una correcta praxi de desenvolupament segons els que hem après a les assignatures de base de dades cursades al llarg del grau.

Pel que respecta a aquest últim punt i en referència a les restriccions textuals, veiem que una d'elles l'hem pogut resoldre amb una restricció del tipus "CHECK". Cal recordar que la clàusula "CHECK" es fa servir per fer comprovacions entre columnes (atributs) de la mateixa taula (entitat). En aquest sentit, la restricció textual número 4 la qual té en compte que "la data/hora de registre d'un usuari al sistema ha de ser necessàriament menor a la data/hora de baixa pel mateix usuari en cas que aquesta última no sigui nul·la" es pot resoldre amb una comprovació check perquè com veiem aquesta comprovació implica control sobre 2 atributs de la mateixa entitat, en concret la data del registre de l'usuari i la data de la baixa d'aquest.

Pel que fa a la resta de restriccions textuals observem que aquestes implica control sobre atributs de diferents entitats, de manera que no poden ser resolts amb aquestes comprovacions i caldrà una comprovació amb una sentència SQL que apliqui sobre les entitats pertinents. Aquestes comprovacions es faran en forma de consultes que es cridaran als procediments d'alta/baixa/modificació pertinents.

5. Procediments

Un cop creades les taules a la base de dades, ha arribat el moment de crear els procediments que s'encarregaran d'inserir, eliminar i modificar les dades que la poblaran.

Cal ser conscients que aquests procediments faran quelcom més que simplement inserir, eliminar i modificar-ne les dades. Aquestes tasques addicionals són:

- Dotar a la futura aplicació de procediments als quals cridar per gestionar les dades a la base de dades.
- Lligat amb el punt anterior, evitar que els programadors es vegin en l'obligació de manipular les dades a través de crides SQL "pures" entenent com a tals INSERT's, DELETE's, SELECT's, etc.
- Incloure els controls necessaris a efectuar sobre les dades com, per exemple, les restriccions textuais.
- Incloure una gestió dels possibles errors i excepcions que es puguin ocasionar en l'execució d'aquests procediments i informar degudament a l'usuari que els executi.
- En el proper punt del treball aquests procediments també inclouran les crides als procediments que poblaran les taules del magatzem i que s'encarregaran d'inserir i actualitzar-ne la informació que aquestes contindran.

Es programaran tots aquells procediments que es considerin necessaris per a la gestió de les entitats que componen la nostra base de dades. Cal recalcar que aquestes funcions seran actualitzades posteriorment al treball quan ens encarreguem del magatzem de dades per incloure-hi les funcions del magatzem encarregades d'actualitzar les dades que es troben a les taules que el componen.

Es dividiran els procediments en funció de les entitats sobre les quals actuen. Per cadascuna d'elles es mostraran els procediments creats, els paràmetres que reben, el paràmetre de sortida, les accions que efectua el procediment i les excepcions que tracta.

Abans de passar a veure els procediments cal tenir en compte que aquest comparteixen un seguit de factors que denotarem a continuació i estaran presents en tots ells, de manera que aquests s'obviaran a les explicacions concretes per a cada procediment. Aquests factors són:

- Les excepcions tractades amb missatges concrets per a cadascun dels procediments són aquelles més susceptibles de donar-se en les accions que executem. La resta d'excepcions (potencialment poden ser centenars) es tracten amb un missatge genèric per l'usuari.

- Sempre que es pugui capturar les excepcions que capturi el sistema a través dels seus codis d'error. En els casos de les excepcions que s'han de generar per les restriccions textuais, aquestes es programaran i es llançaran manualment.
- Totes les funcions d'alta reben tots els atributs de l'element a inserir. Per aquestes funcions sempre es controlaran, com a mínim, les excepcions de violació de clau primària (l'element ja existeix), de violació de constraint not null (valors nuls a atributs no permesos), i possibles violacions de claus foranes (atributs referenciats d'altres taules que no existeixen a aquestes).
- Totes les funcions de baixa rebran només l'atribut que identifica a l'element que es vol eliminar. Aquest atribut és suficient per a gestionar-ne la baixa. Pel que respecta a les excepcions de les baixes es controlaran, com a mínim, que l'element que es vol eliminar no estigui referenciat a alguna altra taula per registres existents, de manera que l'eliminació de l'element generi que aquells registres quedin amb una clau forana "orfe". Cal denotar que intentar eliminar una relació que no existeix en una taula donada no genera cap excepció.
- Totes les funcions de modificació d'algun atribut de l'entitat en qüestió reben sempre l'atribut que es vol modificar i el(s) atribut(s) que identifiquen l'element que es vol modificar. Respecte a les excepcions, com a mínim sempre es controlarà que l'atribut que es vol modificar tingui algun valor en cas que aquest no pugui ser nul, així com el fet que el valor de l'atribut existeixi a la taula pertinent si aquest està referenciat (clau forana). Igual que en el cas de les baixes, intentar modificar un element que no existeix a una taula no genera cap excepció.
- Totes les funcions tenen el paràmetre de sortida RSP que retorna un "OK" en cas que l'acció s'hagi executat correctament i, en cas contrari, un missatge d'error especificant-ne l'origen en els casos tractats.
- Totes les funcions, amb independència de les accions que generin a les taules sobre les quals actuen, també fan una inserció a la taula Log per a poder portar un registre de les funcions cridades, els paràmetres d'entrada i sortida i el retorn d'aquestes.
- En tots els casos en els que existeix un paràmetre de tipus timestamp, aquest mai es passarà des de l'exterior si no que es recollirà del sistema dins el procediment.

Cal recordar que aquestes funcions en aquest punt estan incompletes (falta el bloc de les crides a les funcions d'actualització de les taules del data warehouse). Es pot trobar la codificació final de les funcions a l'[Annex](#).

5.1 País

País té dos procediments programades: alta i baixa. Cal recordar que país es una entitat amb un sol atribut, que correspon al seu nom. Per tant atès que aquest es la seva clau primària, no té sentit cap procediment de modificació.

Procediment	País_alta	País_baixa
Paràmetres IN	- nom_pais	- nom_pais
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a País - Inserció a Log	- Eliminació de País - Inserció a Log
Excepcions tractades	- El pais ja existeix - S'ha inserit un pais nul	- Hi ha alguna seu ubicada a aquest pais

5.2 Seu

Aquesta relació té cinc procediments: alta, baixa i un procediment de modificació de la tipologia (no té sentit modificar atributs que formen part de la clau primària ni aquells que la defineixen físicament com el país o la direcció).

Procediment	Seu_alta	Seu_baixa
Paràmetres IN	- codi_seu - direccio_seu - tipologia_seu - pais_seu	- codi_seu
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Seu - Inserció a Log	- Eliminació de Seu - Inserció a Log
Excepcions tractades	- La seu ja existeix - S'ha inserit algun valor nul no permès - El pais no existeix a la taula Pais	- Hi ha algun usuari ubicat a aquesta seu

Procediment	Seu_modif_tipologia
Paràmetres IN	- codi_seu - tipologia_seu
Paràmetres OUT	- RSP
Accions executades	- Update taula Seu - Inserció a Log
Excepcions tractades	

5.3 Usuari

Els procediments d'aquesta relació mereixen una menció especial. En concret això és degut al fet que en aquest cas quan un usuari es dona de baixa, aquest no s'elimina de la base de dades, sinó que el que es fa és una actualització a la taula on es dona valor a l'atribut de la data de baixa per a aquell usuari. D'aquesta manera, el procediment de usuari_baixa no efectuarà un delete sinó un Update per a la data de baixa.

Finalment també hem creat una funció per modificar la seu a la qual pertany l'usuari, que és l'únic atribut que no forma part de la clau primària ni és un timestamp. Sent així, les funcions quedarien de la següent manera:

Procediment	usuari_alta	usuari_baixa
Paràmetres IN	- mail_usuari - seu_usuari	- mail_usuari
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Usuari - Inserció a Log	- Update taula Usuari - Inserció a Log
Excepcions tractades	- L'usuari ja existeix - S'ha inserit algun valor nul no permès - La seu no existeix a la taula Seu - El valor de la data de baixa és inferior a la data de alta	

Procediment	usuari_modif_seu
Paràmetres IN	<ul style="list-style-type: none"> - Mail_usuari - Seu_usuari
Paràmetres OUT	<ul style="list-style-type: none"> - RSP
Accions executades	<ul style="list-style-type: none"> - Update taula Usuari - Inserció a Log
Excepcions tractades	<ul style="list-style-type: none"> - La seu és nula - La seu no existeix a la taula Seu

5.4 Interacció

Interacció té dues funcions programades: alta i baixa. Recordem que aquesta entitat és un registre de les interaccions al sistema per part dels usuaris de manera que no hi ha sentit a modificar cap atribut d'una entrada ja registrada així com tampoc eliminar-ne cap interacció ja que al final es tracta d'un històric.

Procediment	interaccio_alta
Paràmetres IN	<ul style="list-style-type: none"> - Usuari_interaccio - Accio
Paràmetres OUT	<ul style="list-style-type: none"> - RSP
Accions executades	<ul style="list-style-type: none"> - Inserció a Interacció - Inserció a Log
Excepcions tractades	<ul style="list-style-type: none"> - Linteracció ja existeix - S'ha inserit algun valor nul no permès

5.5 NivellSuport

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediments de modificació.

Procediment	nivellsuport_alta	nivellsuport_baixa
Paràmetres IN	- Nivell_nivellsuport - Valor_nivellsuport	- Nivell_nivellsuport
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a NivellSuport - Inserció a Log	- Eliminació NivellSuport - Inserció a Log
Excepcions tractades	- El nivell ja existeix - S'ha inserit un nivell nul	- Hi ha algun grup de suport al que correspon aquest nivell

5.6 GrupSuport

Relació amb quatre procediments: alta, baixa i un procediment de modificació pel nom del grup.

Procediment	grupsuport_alta	grupsuport_baixa
Paràmetres IN	- Codi_grupsuport - Nom_grupsuport - Nivell_grupsuport	- Codi_grupsuport
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a GrupSuport - Inserció a Log	- Eliminació GrupSuport - Inserció a Log
Excepcions tractades	- El grup ja existeix - S'ha inserit algun valor nul no permès - El nivell no existeix a la taula NivellSuport	- Aquest grup està relacionat amb algun ticket o servei existent

Procediment	Grupsuport_modif_nom
Paràmetres IN	- Codi_grupsuport Nom_grupsuport
Paràmetres OUT	RSP
Accions executades	- Update taula GrupSuport Inserció a Log
Excepcions tractades	La direcció és nula

5.7 Tècnic

Aquesta relació té tres procediments: alta, baixa i una funció de modificació del grup on està assignat, que és l'únic atribut que no forma part de la clau primària

Procediment	tecnic_alta	tecnic_baixa
Paràmetres IN	- Mail_tecnic - Grup_tecnic - Nivell_grupsuport	- Mail_tecnic
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Tecnic - Inserció a Log	- Eliminació de Tecnic - Inserció a Log
Excepcions tractades	- El tecnic ja existeix - S'ha inserit algun valor nul no permès - El grup no existeix a la taula GrupSuport	- Aquest grup està relacionat amb algun canvi d'estat d'un ticket existent

Procediment	tecnic_modif_grup
Paràmetres IN	- Mail_tecnic - Grup_tecnic
Paràmetres OUT	- RSP
Accions executades	- Update taula Tecnic - Inserció a Log
Excepcions tractades	- El grup és nula - El grup no existeix a la taula GrupSuport

5.8 Prioritat

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	Prioritat_alta	Prioritat_baixa
Paràmetres IN	- Nivell_prioritat	- nivell_prioritat
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Prioritat - Inserció a Log	- Eliminació de Prioritat - Inserció a Log
Excepcions tractades	- La prioritat ja existeix - S'ha inserit una prioritat nula	- Hi ha algun tikket o servei amb aquesta prioritat

5.9 GrupServei

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	grupservei_alta	grupservei_baixa
Paràmetres IN	- grup_grupservei	- grup_grupservei
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció GrupServei ^a - Inserció a Log	- Eliminació de GrupServei - Inserció a Log
Excepcions tractades	- La grup ja existeix - S'ha inserit un grup de serveis nul	- Hi ha algun grup de suport que ofereix aquest grup de serveis

5.10 Servei

Relació amb sis procediments: alta, baixa i quatre procediments de modificació pels quatre atributs que no formen part de la clau primària.

Procediment	servei_alta	servei_baixa
Paràmetres IN	- Codi_servei - descripcio_servei - grup_servei - temps_mig_servei - prioritat_servei	- codi_servei
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Servei - Inserció a Log	- Eliminació de Servei - Inserció a Log
Excepcions tractades	- El servei ja existeix - S'ha inserit algun valor nul no permès - El grup de servei no existeix a la taula GrupServei - La prioritat no existeix a la taula Prioritat	- Aquest servei està relacionat amb algun ticket existent

Procediment	servei_modif_descripcio	servei_modif_grup
Paràmetres IN	- Codi_servei - descripcio_servei	- codi_servei - grup_servei
Paràmetres OUT	- RSP	- RSP
Accions executades	- Update taula Servei - Inserció a Log	- Update taula Servei - Inserció a Log
Excepcions tractades		- El grup es nul - El grup no existeix a la taula GrupServei

Procediment	servei_modif_temps	servei_modif_prioritat
Paràmetres IN	- Codi_servei - Temps_mig_servei	- codi_servei - prioritat_servei
Paràmetres OUT	- RSP	- RSP
Accions executades	- Update taula Servei - Inserció a Log	- Update taula Servei - Inserció a Log
Excepcions tractades	- El temps mig es nul	- La prioritat es nula - La prioritat no existeix a la taula Prioritat

5.11 OfertaGrupServei

Relació amb dos procediments: alta i baixa. Els dos atributs d'aquesta relació formen part de la seva clau primària pel que no és necessari cap procediment de modificació.

Procediment	ofertagrupservei_alta	ofertagrupservei_baixa
Paràmetres IN	- grup_ofertagrupservei - servei_ofertagrupservei	- grup_ofertagrupservei - servei_ofertagrupservei
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a OfertaGrupServei - Inserció a Log	- Eliminació de OfertaGrupServei - Inserció a Log
Excepcions tractades	- La oferta ja existeix - S'ha inserit algun valor nul no permès	- El servei no existeix - El grup no existeix

5.12 TipusTicket

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	tipusticket_alta	tipusticket _baixa
Paràmetres IN	- Tipus_tipusticket	- Tipus_tipusticket
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a TipusTicket - Inserció a Log	- Eliminació de TipusTicket - Inserció a Log
Excepcions tractades	- El tipus de ticket ja existeix - S'ha inserit un típus nul	- Hi ha algun ticket al que correspon aquest típus

5.13 Canal

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	canal_alta	canal _baixa
Paràmetres IN	- Tipus_canal	- Tipus_canal
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Canal - Inserció a Log	- Eliminació de Canal - Inserció a Log
Excepcions tractades	- El canal ja existeix - S'ha inserit un canal nul	- Hi ha algun ticket al que correspon aquest canal

5.14 CreacioTicket

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	creacioticket_alta	creacioticket _baixa
Paràmetres IN	- Metode_creacioticket	- Metode_creacioticket
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a CreacioTicket - Inserció a Log	- Eliminació de CreacioTicket - Inserció a Log
Excepcions tractades	- El metode de creació ja existeix - S'ha inserit un metode nul	- Hi ha algun ticket al que correspon aquest metode de creació

5.15 Ticket

Relació amb quatre funcions: alta, baixa i dues funcions de modificació. Cal veure que encara que aquesta relació tingui molts atributs, no té sentit poder-los modificar tots. Per exemple, modificar l'usuari que ha generat el ticket, el servei que sol·licita, el tipus de ticket... seria en realitat modificar quelcom tan substancial que equivaldria a un ticket nou. Encara que aquests elements no en siguin claus primàries ni identifiquin al ticket inequívocament. D'aquesta manera només es podran modificar els atributs canal i creació.

Procediment	ticket_alta	ticket_baixa
Paràmetres IN	<ul style="list-style-type: none"> - Numero_ticket - Servei_ticket - Prioritat_ticket - Tipus_ticket - Canal_ticket - Creacio_ticket - usuari_ticket 	<ul style="list-style-type: none"> - numero_ticket
Paràmetres OUT	<ul style="list-style-type: none"> - RSP 	<ul style="list-style-type: none"> - RSP
Accions executades	<ul style="list-style-type: none"> - Inserció a Ticket - Inserció a Log 	<ul style="list-style-type: none"> - Eliminació de Ticket - Inserció a Log
Excepcions tractades	<ul style="list-style-type: none"> - El ticket ja existeix - S'ha inserit un mètode nul - Algun dels elements referenciats no existeixen a les seves taules 	<ul style="list-style-type: none"> - El ticket està relacionat amb algun canvi d'esat o grup de suport existent

Procediment	ticket_modif_creacio	ticket_modif_canal
Paràmetres IN	- Numero_ticket - Creacio_ticket	- Numero_ticket - Canal_ticket
Paràmetres OUT	- RSP	- RSP
Accions executades	- Update taula Ticket - Inserció a Log	- Update taula Ticket - Inserció a Log
Excepcions tractades	- El metode és nul - El mètode no existeix a la taula CreacioTicket	- El canal és nul - El canal no existeix a la taula Canal

5.16 AssigTicketGrup

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	assigticketgrup_alta	assigticketgrup _baixa
Paràmetres IN	- Ticket_assigticketgrup - Grup_assigticketgrup	- Ticket_assigticketgrup - Grup_assigticketgrup
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a AssigTicketGrup - Inserció a Log	- Eliminació de AssigTicketGrup - Inserció a Log
Excepcions tractades	- L'assignació ja existeix - S'ha inserit algun valor nul no permès - El grup no existeix a la taula GrupSuport - El ticket no existeix a la taula Ticket - El nivell del grup és inferior al nivell del grup actual - El grup no ofereix el servei que el ticket requereix	

En aquest cas ens trobem que hem agut de gestionar dues restriccions textuais. Per fer-ho no hi ha suficient amb les excepcions que oracle té programades i el que hem fet ha sigut fer les comprovacions a les taules pertinents i en cas que la restricció es violés fer saltar una excepció de manera manual.

```

---
--Restriccions textuais
--nivell del grup que es vol assignar
SELECT nivellsuport.valor INTO nivell_futur
FROM grupsuport INNER JOIN nivellsuport ON nivellsuport.nivell = grupsuport.nivell
WHERE grupsuport.codi = grup_assigticketgrup;

--nivell del grup que té actualment assignat
SELECT MAX (nivellsuport.valor) INTO nivell_actual --agafem el nivell del grup assignat actualment al ticket
FROM assigticketgrup INNER JOIN grupsuport ON assigticketgrup.grup = grupsuport.codi
INNER JOIN nivellsuport ON grupsuport.nivell = nivellsuport.nivell
WHERE assigticketgrup.ticket = ticket assigticketgrup
ORDER BY assigticketgrup.data_hora DESC;

IF nivell_futur < nivell_actual THEN
    RAISE nivell_decreixent;
END IF;

--servei que sol·licita el ticket
SELECT COUNT (*) INTO oferta_servei
FROM ticket
WHERE numero = ticket_assigticketgrup AND servei IN (
    SELECT servei
    FROM ofertagrupservei
    WHERE grup = grup_assigticketgrup
);

IF oferta_servei < 1 THEN
    RAISE servei_no_ofert;
END IF;

```

Tal com es pot veure, pel cas de la primera restricció el que fem és agafar el nivell del grup que es vol assignar (paràmetre d'entrada) i el nivell del grup que té assignat el tiquet actualment (de les assignacions per aquell tiquet, aquella que tingui la data_hora més alta). Un cop tenim els dos nivells només resta comprovar que el nivell del grup que es vol assignar és com a mínim igual o superior al nivell del grup assignat actualment. En cas contrari, amb la clàusula RAISE aixequem de manera manual l'excepció que hem declarat prèviament.

En el cas de la segona restricció, només cal comprovar si el servei que sol·licita el tiquet és ofert pel grup de suport que s'està intentant assignar. De la mateixa manera, si aquest no és el cas, s'aixeca l'excepció.

Finalment al bloc d'excepcions es tracten aquestes de manera diferenciada i finalment com sempre tractem la resta en funció dels codis d'error que oracle ens reporti:

```

--Tractament Excepcions
EXCEPTION
WHEN nivell_decreixent THEN
    RSP := 'ERROR: Restricció textual. El nivell del grup que s'intenta assignar al ticket és inferior al nivell del grup que el ticket ja té assignat actualment';

WHEN servei_no_ofert THEN
    RSP := 'ERROR: Restricció textual. El grup que s'intenta assignar al ticket no ofereix el servei que el ticket demana';

WHEN OTHERS THEN
    IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquesta assignació ja existeix';
    ELSIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
    ELSIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. El grup o el ticket no existeixen';
    ELSE RSP := 'ERROR: Error inesperat';
    END IF;

```

5.17 EstatTicket

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	estatticket_alta	estatticket_baixa
Paràmetres IN	- estat_estatticket	- estat_estatticket
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a EstatTicket - Inserció a Log	- Eliminació de EstatTicket - Inserció a Log
Excepcions tractades	- L'estat ja existeix - S'ha inserit un estat nul	- Hi ha algun ticket al que correspon aquest estat

5.18 CanviEstat

Aquesta relació consta només d'un procediment: alta. El fet que no consti de cap procediment de baixa és que no té sentit perdre elements de la relació canvi estat, atès que aquesta relació s'entén com un històric dels diferents estats pels quals ha passat un ticket. És a dir, un canvi d'estat no és una modificació d'una entrada de la taula sinó una nova entrada, generant en la seva totalitat l'històric.

Procediment	canviestat_alta
Paràmetres IN	- Ticket_canviestat - Tecnic_canviestat - Estat_canviestat
Paràmetres OUT	- RSP
Accions executades	- Inserció a CanviEstat - Inserció a Log
Excepcions tractades	- El canvi d'estat ja existeix - S'ha inserit algun valor nul no permès - El tècnic no pertany al grup assignat al ticket actualment

De la mateixa manera que ja passava a AssigTicketGrup en aquesta funció es controla la restricció textual que dicta que un canvi d'estat l'ha de generar un tècnic que correspongui al grup assignat al ticket. Aquest es tracta de la mateixa manera que hem exposat a alta_assigticketgrup.

5.19 EstatSms

Relació amb dos procediments: alta i baixa. Té un únic atribut pel que no és necessari cap procediment de modificació.

Procediment	estatsms_alta	estatsms_baixa
Paràmetres IN	- Estat_estatsms	- Estat_estatsms
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a EstatSms - Inserció a Log	- Eliminació de EstatSms - Inserció a Log
Excepcions tractades	- L'estat ja existeix - S'ha inserit un estat nul	- Hi ha algun missatge al que correspon aquest estat

5.20 Missatge

Aquesta relació té dos procediments: alta i una funció de modificació de l'estat del sms. Els altres atributs que no formen part de la clau primària són l'emissor i el ticket al que està associat el missatge, atributs que no té sentit modificar. Per una altra banda en el cas dels missatges tampoc té sentit la funció de baixa atès que perdríem el factor històric

Procediment	missatge_alta	missatge_modif_estat
Paràmetres IN	- Codi_missatge - emissor_missatge - estat_missatge - ticket_missatge	- Estat_missatge
Paràmetres OUT	- RSP	- RSP
Accions executades	- Inserció a Missatge - Inserció a Log	- Update de Missatge - Inserció a Log
Excepcions tractades	- El missatge ja existeix - S'ha inserit algun valor nul no permès - L'emissor del missatge ha de ser 'SYS' o l'usuari al que pertany el ticket associat	- L'estat és nul - L'estat no existeix a la taula EstatSms

En la funció d'alta controlem la darrera de les restriccions textuais que tenim a la nostra base de dades: L'usuari emissor del missatge ha de ser 'SYS' (usuari que representa el sistema) o aquell usuari que ha generat el ticket associat al missatge en qüestió.

5.21 Log

Recordem que aquesta entitat és un registre que guarda un històric de totes les crides a les funcions anteriors, desant els seus paràmetres d'entrada i sortida. En aquesta taula només té sentit efectuar insercions, ni modificacions ni baixes, ja que perdiem la veracitat de l'històric. A més, les crides d'inserció estan encastades dins de totes les funcions anteriors, de manera que no hi ha necessitat de generar cap procediment, ja que en cap cas un usuari posterior d'aquesta base de dades (programador de l'aplicació, usuari final...) n'hauran de fer ús.

6. Data Warehouse

En aquest apartat de la memòria veurem com es planifica el magatzem de dades. Aquest magatzem ha de ser capaç de donar resposta a certes consultes en un temps constant 1, és a dir, la consulta no pot tenir cap funció d'agregats que faci que el temps variï en funció de la quantitat de registres que contingui la taula on busquem les dades per a donar resposta a la consulta.

Per tant, el primer que haurem de fer és analitzar aquestes consultes per a determinar quines dades necessitem per a fer possible el càlcul de la resposta. Un cop esclarides les dades necessàries, haurem de construir les estructures pertinents per a poder emmagatzemar-les així com els processos necessaris per a poder actualitzar-les degudament.

Finalment ja només restarà crear els processos de consulta sobre aquestes noves taules que generin una resposta a temps constant.

6.1 Anàlisi de les dades requerides

Tal com ja hem exposat, les dades requerides estan en funció de les consultes que es vol efectuar sobre aquest magatzem de dades. És per això que per determinar-ne les dades, el primer que farem serà analitzar aquestes consultes.

- **C1:** *Donat un mes qualsevol, temps mig de resolució d'incidències per cada tipus de prioritat.*

Per aquesta consulta ens caldrà desar en una taula, per a més i per a cada prioritat dels tickets de tipus incidència, una suma total del temps que s'ha trigat a resoldre'ls i el nombre total d'incidències resoltes. D'aquesta manera al moment de consultar la taula només caldrà efectuar una divisió per a poder donar el resultat en temps constant 1.

Paràmetres IN: més

- **C2:** *Donats un país i un mes concrets, percentatge de peticions resoltes per sobre del temps definit (tenint en compte totes les prioritats i el seu diferent temps de resolució).*

En aquest cas ens caldrà desar els país, el més i per a cada parell d'aquestes el nombre total d'incidències resoltes per sobre del temps definit així com la suma total d'aquests temps.

Paràmetres IN: més, país

- **C3:** *Usuari que, en el moment de realitzar la consulta, ha creat més tickets de tipus incidències.*

Aquí ens cal, per a cada usuari desar el nombre de tiquets de tipus incidència que ha creat.

Paràmetres IN: -

- **C4:** *Top10 per seu de les peticiones més sol·licitades (tenint en compte totes les dades de que es disposi).*

En aquest cas cal guardar, per a cada seu i per a cada servei diferent associat a un tiquet de tipus petició, el nombre de vegades que aquest s'ha sol·licitat.

Paràmetres IN: seu

- **C5:** *Percentatge d'increment (o decrement) del número de tickets resolts respecte al mes anterior.*

Tal com veiem, aquí ens caldrà desar, per a cada mes, el nombre total de tiquets resolts. A l'hora de la consulta només caldrà calcular el percentatge del mes actual respecte a l'anterior.

Paràmetres IN: -

- **C6:** *Entrada del catàleg de serveis que ha sigut més sol·licitada. S'haurà de tenir en compte només els tickets de tipus petició.*

Cal desar, per a cada servei existent, el nombre de vegades que aquest s'ha sol·licitat en un tiquet de tipus petició.

Paràmetres IN: -

- **C7:** *Seu en la que, en l'any en curs, s'han creat un número major d'incidències.*

Per a poder donar resposta a aquesta consulta cal desar, per cada any i per cada seu, el nombre d'incidències creades.

Paràmetres IN: -

- **C8:** *Donat un any concret, hores totals invertides en resolució de tickets pel nivell L3.*

En aquest cas, caldrà desar, per a cada any, la suma total d'hores invertides en tiquets assignats a grups de suport amb nivell L3.

Paràmetres IN: any

- **C9:** *Número d'incidències no resoltes pels 3 primers nivells de suport (L1- L3) i assignades al grup de Problem Management en l'any en curs.*

Cal desar, per a cada any, el número d'incidències assignades a un grup de suport de nivell 'Problem Management'.

Paràmetres IN: -

- **C10:** *En l'any en curs, percentatge de missatges que no han sigut respostos. Es considerarà que un missatge no ha sigut respost quan, per un determinat ticket, només hi ha missatges enviats pel sistema.*

En aquest cas caldrà desar per a cada any el nombre de tiquets totals existents al sistema així com el nombre de tiquets existents al sistema que no han sigut respostos. A l'hora de donar la sortida només caldrà fer la divisió entre els 2 valors.

S'ha determinat amb el consultor que sempre es pot considerar que un tiquet tindrà, com a mínim un missatge enviat pel sistema. D'aquesta manera, només caldrà afegir un nou tiquet al total de tiquets i al total de tiquets no respostos a l'hora, i decrementar el nombre de tiquets no respostos quan es generi el primer missatge associat a un tiquet on l'emissor no sigui el sistema.

Paràmetres IN: -

- **C11:** *En un moment concret, número total de tickets no oberts. Es considera obert quan el seu estat no es ni Resolt ni Cancel·lat.*

Aquí només ens cal un valor i és el nombre de tiquets tancats, és a dir, un valor que s'incrementarà cada cop que un tiquet passi a un estat de 'Resolt' o 'Cancel·lat'.

Paràmetres IN: -

- **C12:** *En un mes i un país concrets, percentatge de tickets creats per cada un dels diferents canals d'entrada (telèfon, xat o web).*

Finalment per a donar resposta a aquesta consulta ens caldrà desar, per a cada combinació de mes, país i canal d'entrada, el total de tiquets creats i la suma de tiquets creats per aquell canal en qüestió.

D'aquesta manera a l'hora de donar el resultat només caldrà dividir el nombre total de tiquets d'aquell país, per aquell mes respecte als creats per aquell país i mes però pel canal donat.

Paràmetres IN: país, més

6.2 Estructura i creació de les taules

Ara que ja ens hem fet a la idea de les dades que hem de desar per a poder donar resposta a totes les consultes que necessitem, passem a formalitzar-ho per a definir l'estructura de cadascuna de les taules i posteriorment a traduir-ho a sentències de creació en SQL. Atès que no existeix cap relació entre les taules que conformen el magatzem, procedirem a fer un disseny lògic seguit de la traducció física per a cada taula relacionada a cadascuna de les consultes vistes.

Abans però cal deixar clar alguns conceptes:

- En aquelles consultes que requereixen que pel més actual es doni una dada, possiblement n'hi hauria prou amb desar i actualitzar-les dades corresponents al més actual a la taula del DW, això no obstant, he considerat que el DW té una definició i utilitat d'historial de manera que no s'eliminaran les dades de mesos anteriors, creant noves entrades per a cada mes nou si és necessari.
- Per aquelles taules en les quals el mes és un atribut, també es desarà l'any ('ani', ja que 'any' és una paraula reservada), atès que el més (entès amb un valor numèric de rang [1..12]) no és prou significatiu.
- Els temps sempre seran expressats en segons enters. Serà responsabilitat de l'aplicació gestionar el format que se'n vulgui donar.
- Per a cada taula, veurem el seu model lògic i en farem una explicació mes formal dels atributs que desa. La traducció física (sentències SQL) es poden trobar a l'[Annex](#).
- Podem observar que hi ha certes consultes que no reben com a paràmetre l'any sinó que es vol obtenir una dada concreta per l'any actual. No obstant hem decidit guardar a la taula un històric per a cada any atès que això suposa un augment del volum de dades irrisori i pot servir per un històric de caràcter BI en un futur.

1. DW_temps_incidencies_prio

```
DW_temps_mig_incidencies_prioritat  
(mes, ani, prioritat, qtt_incidencies, temps_total)  
    {mes, any, prioritat} is PRIMARY KEY  
    {qtt_incidencies} is NOT NULL  
    {temps_total} is NOT NULL
```

Tal com podem observar, aquesta taula desa els atributs temporals mes i any, la prioritat i per a cada tripleta en desa la quantitat d'incidències i el temps total de resolució.

2. DW_peticions_sobre_temps

DW_peticions_resoltes_sobre_temps
(mes, ani, pais, **qtt_peticions**, **qtt_peticions_sobre_temps**)
 {mes, ani, pais} is PRIMARY KEY
 {qtt_peticions} is NOT NULL
 {qtt_peticions_sobre_temps } is NOT NULL

En aquest cas, la taula desa els atributs temporals mes/any i el país. Per a cada tripleta en guarda la quantitat de peticions totals i la quantitat de peticions resoltes en un temps superior al temps mig previst.

3. DW_usuari_mes_incidencies

DW_usuari_mes_incidencies
(usuari, **qtt_incidencies**)
 {usuari} is PRIMARY KEY
 {qtt_incidencies} is NOT NULL

Aquesta taula desa, per a cada usuari, la quantitat d'incidències que ha creat.

4. DW_top10_peticions_seu

DW_top10_peticions_seu
(seu, servei, **qtt_peticions**)
 {seu, servei} is PRIMARY KEY
 {qtt_peticions} is NOT NULL

En aquest cas es desa per a cada seu i servei, la quantitat de peticions.

5. DW_tickets_resolts_mes

DW_tickets_resolts_mes
(mes, ani, **qtt_tickets**)
 {mes, ani} is PRIMARY KEY
 {qtt_tickets} is NOT NULL

Per a donar resposta a aquesta consulta, n'hi haurà prou amb què la taula desi els valors temporals mes/any i per a cadascun d'ells la quantitat de tiquets que s'han creat.

6. DW_servei_peticions

DW_servei_peticions
(servei, **qtt_peticions**)
 {servei} is PRIMARY KEY
 {qtt_peticions} is NOT NULL

En aquest cas, només cal desar per a cada servei la quantitat de peticions que s'han fet sol·licitant-lo.

7. DW_incidencies_seu_any

DW_incidencies_seu_any
(ani, seu, **qtt_incidencies**)
 {ani, seu} is PRIMARY KEY
 {qtt_incidencies} is NOT NULL

Aquesta taula desa el valor temporal any, la seu i per a cada combinació dels dos la quantitat d'incidències creades .

8. DW_hores_tickets_L3

DW_hores_tickets_L3
(ani, **qtt_hores**)
 {ani} is PRIMARY KEY
 {qtt_hores} is NOT NULL

En aquest cas només cal desar l'any i la quantitat d'hores invertides per grups de nivell L3 en tickets.

9. DW_incidencies_PM

DW_incidencies_PM
(ani, **qtt_incidencies**)
 {ani} is PRIMARY KEY
 {qtt_incidencies} is NOT NULL

Aquesta taula desa el valor temporal any i la quantitat d'incidències creades per aquell any.

10. DW_sms_no_respostos

(ani, **qtt_tickets**, **qtt_no_respostos**)
 {ani} is PRIMARY KEY
 {qtt_tickets} is NOT NULL
 {qtt_no_respostos} is NOT NULL

Podem observar que en aquest cas desem l'any i per a cada any la quantitat de tickets creats i la quantitat de tickets no respostos.

11.DW_tickets_tancats

DW_tickets_tancats
(qtt_tancats)
{qtt_tancats} is PRIMARY KEY

Sens dubte una de les taules més bàsiques, atès que la informació que ha de proporcionar és la de la quantitat de tickets cancel·lats o resolts.

12.DW_tickets_per_canal

DW_tickets_per_canal
(pais, mes, any, canal, qtt_tickets_canal, qtt_tickets_total)
{pais, mes, any, canal} is PRIMARY KEY
{qtt_tickets_canal } is NOT NULL
{qtt_tickets_total} is NOT NULL

Finalment aquesta taula desa els atributs temporals mes/any, país, canal i per a cada conjunt dels anteriors la quantitat de tiquets oberts per aquell canal així com la quantitat de tiquets totals.

6.3 Procediments d'actualització

És el torn de generar els procediments d'actualització per a cadascuna de les taules que hem creat per al magatzem. Algunes de les taules només necessitaran un procediment que s'encarregui d'incrementar un cert valor donats uns paràmetres mentre que d'altres també necessitaran procediments per decrementar-los en cas que es donin certes situacions. Cal recordar que aquests procediments seran cridats dins dels procediments que hem generat prèviament per gestionar les taules de la base de dades operativa. D'aquesta manera, caldrà generar aquests nous procediments per al magatzem però també acabar de complimentar els procediments de les taules de la base de dades operativa per encastar-hi, en cada cas, les crides a aquests procediments.

És important veure també que aquests procediments seran aquells que hauran d'actualitzar les dades de les taules del magatzem però potencialment també hauran de crear les entrades per primer cop si aquestes no existeixen. Per aquesta raó, en el cas dels procediments d'agregació (o increment) sempre es troba un control en el qual es fa un Update i en cas que aquest no hagi modificat cap entrada, es fa l'insert a la taula.

Per contra en el cas dels procediments de decrement (en els que s'hi ha de decrementar algun valor a causa que algun tiquet s'ha eliminat del sistema, s'ha cancel·lat o altres) no cal fer cap control atès que quan es crida algun procediment de la base de dades per fer alguna d'aquestes accions de baixa podem estar segurs que aquests elements s'han donat d'alta prèviament. De no ser així saltaria l'excepció pertinent codificada als procediments de les taules de la base de dades operatives i no suposaria major problema.

Com ja és costum, la codificació dels procediments es pot trobar a l'[Annex](#) de manera que el que farem a continuació és veure la quantitat de procediments que existiran per taula, els paràmetres d'entrada que reben, una explicació de la seva tasca i aquells procediments d'ABM de les taules de la base de dades operativa des d'on es llençarà aquests per anar actualitzant degudament el magatzem.

6.3.1 DW_temps_incidencies_prio

❖ DW_temps_incidencies_prio_INC

Quan un tiquet de tipus incidència passi al estat 'resolt' per a la prioritat d'aquell tiquet s'incrementarà la quantitat d'incidències resoltes i es sumarà el temps transcorregut des de la seva creació fins al moment del canvi d'estat a 'resolt' al total de temps ja computat en la resolució de incidències. Recordem que un canvi d'estat es tradueix a la base de dades com una alta a la taula CanviEstat.

Serà executat des dels següents procediments:

- **canviestat_alta** (quan tel tipus de tiquet sigui incidència, i l'estat passi a ser resolt)

Paràmetres d'entrada:

- prioritat del tiquet
- data de creació del tiquet

6.3. DW_peticions_sobre_temps

❖ DW_peticions_sobre_temps_INC

Quan un tiquet de tipus petició passi a l'estat 'resolt' s'ha de sumar una unitat al nombre de peticions resoltes per aquell mes/país. Si a més el temps que ha trigat a resoldre's des de la seva creació és superior al temps mitjà de resolució per al servei que sol·licita el tiquet, s'ha de sumar una unitat a la quantitat de tiquets resolts per sobre del temps mitjà per aquell país/any. Recordem que un canvi d'estat es tradueix a la base de dades com una alta a la taula CanviEstat.

Serà executat des dels següents procediments:

- **canviestat_alta** (quan tel tipus de tiquet sigui petició, i l'estat passi a ser resolt)

Paràmetres d'entrada:

- temps mig de resolució pel servei que sol·licita el tiquet
- data de creació del tiquet
- país de l'usuari que ha creat el tiquet

6.3.3 DW_usuari_mes_incidencies

❖ DW_usuari_mes_incidencies_INC

Quan un usuari generi un nou tiquet de tipus incidència, s'ha de sumar el nombre d'incidències creades per aquell usuari.

Serà executat des dels següents procediments:

- **ticket_alta** (quan tel tipus de tiquet sigui incidència)

Paràmetres d'entrada:

- usuari que ha creat el tiquet

❖ DW_usuari_mes_incidencies_DEC

Aquest cas es donarà quan un tiquets de tipus incidència sigui eliminat del sistema o el seu estat passi a 'cancel·lat'

Serà executat des dels següents procediments:

- **ticket_baixa** (quan tel tipus de tiquet sigui incidència)
- **canviestat_alta** (quan tel tipus de tiquet sigui incidència, i l'estat passi a ser 'cancel·lat')

Paràmetres d'entrada:

- usuari que ha creat el ticket

6.3.4 DW_top10_peticions_seu

❖ DW_top10_peticions_seu_INC

Quan es genera un nou tiquet de tipus petició, s'ha d'incrementar en una unitat la quantitat de peticions sol·licitades per a la seu de l'usuari sol·licitant i pel servei que aquest sol·licita.

Serà executat des dels següents procediments:

- **ticket_alta** (quan el tipus de tiquet sigui petició)

Paràmetres d'entrada:

- seu de l'usuari que crea la petició
- servei que sol·licita l'usuari

❖ DW_top10_peticions_seu_DEC

De la mateixa manera, quan un tiquet de tipus petició s'elimina del sistema o l'usuari que l'ha creat el cancel·la s'ha de decrementar la quantitat de peticions per la seu de l'usuari i pel servei que sol·licitava el tiquet.

Serà executat des dels següents procediments:

- **ticket_baixa** (quan tel tipus de tiquet sigui petició)
- **canviestat_alta** (quan tel tipus de tiquet sigui petició, i l'estat passi a ser 'cancelat')

Paràmetres d'entrada:

- seu de l'usuari que crea la petició
- servei que sol·licita l'usuari

6.3.5 DW_tickets_resolts_mes

❖ DW_tickets_resolts_mes_INC

Cada cop que un tiquet passa a l'estat de 'resolt', s'ha d'incrementar el nombre de tiquets resolts per a aquell mes/any

Serà executat des dels següents procediments:

- **canviestat_alta**

6.3.6 DW_servei_peticions

❖ DW_servei_peticions_INC

Cada cop que es crea un ticket de tipus petició, s'ha d'incrementar el nombre de vegades que s'ha sol·licitat el servei que sol·licita el ticket en qüestió.

Serà executat des dels següents procediments:

- **ticket_alta** (quan tel tipus de tiquet sigui petició)

Paràmetres d'entrada:

- servei que sol·licit

❖ DW_servei_peticions_DEC

De la mateixa manera, cada cop que s'elimini un tiquet de tipus petició o aquest passi a un estat de 'cancel·lat', s'ha de decrementar el nombre de vegades que s'ha sol·licitat el servei que demana el tiquet en qüestió.

Serà executat des dels següents procediments:

- **ticket_baixa**
- **canviestat_alta** (quan el tipus de tiquet sigui petició i l'estat passi a ser 'cancel·lat')

Paràmetres d'entrada:

- servei que sol·licita

6.3.7 DW_incidencies_seu_any

❖ DW_incidencies_seu_any_INC

Cada cop que es genera un nou tiquet de tipus 'incidència', s'ha d'incrementar el nombre d'incidències per aquella seu/any

Serà executat des dels següents procediments:

- **ticket_alta** (quan el tipus de tiquet sigui incidència)

Paràmetres d'entrada:

- seu de l'usuari que genera el tiquet

❖ DW_incidencies_seu_any_DEC

De la mateixa manera, cada cop que s'elimina un tiquet de tipus 'incidència' del sistema o es cancel·la, s'ha de decrementar el nombre d'incidències per aquella seu/any

Serà executat des dels següents procediments:

- **ticket_baixa** (quan el tipus de tiquet sigui incidència)
- **canviestat_alta** (quan el tipus de tiquet sigui incidència, i l'estat passi a ser 'cancel·lat')

Paràmetres d'entrada:

- seu de l'usuari que genera el tiquet

6.3.8 DW_hores_tickets_L3

❖ DW_hores_tickets_L3_INC

Cada cop que un tiquet s'assigna a un grup de suport de nivell "Problem Management", s'ha d'incrementar el temps que el tiquet a passat assignat a un grup de nivell L3 per aquell any. També s'ha de contemplar que cada cop que un tiquet assignat a un grup L3 passi a estat de resolt, s'ha d'incrementar el temps que el tiquet ha passat assignat al grup L3.

Serà executat des dels següents procediments:

- **assigticketgrup_alta** (quan el grup sigui PM)
- **canviestat_alta** (quan el grup sigui L3 i l'estat sigui 'Resolt')

Paràmetres d'entrada:

- data en que el tiquet es va assignar al grup de nivell L3

En aquest cas no té sentit fer un decrement atès que encara que el tiquet es cancel·li o s'elimini del sistema, el temps invertit pel grup de nivell L3 segueix vigent i no es pot eliminar.

6.3.9 DW_incidencies_PM

❖ DW_incidencies_PM_INC

Cada cop que un tiquet de tipus incidència s'assigna a un grup de suport de nivell "Problem Management", s'ha d'incrementar el nombre d'incidències assignades a PM per aquell any.

Serà executat des dels següents procediments:

- **assigticketgrup_alta**

❖ DW_incidencies_PM_DEC

De la mateixa manera, cada cop que s'elimina una assignació d'un tiquet un grup PM s'ha de decrementar el nombre d'incidències assignades a PM per aquell any.

Serà executat des dels següents procediments:

- **assigticketgrup_baixa**

6.3.10 DW_sms_no_respostos

❖ DW_sms_no_respostos_INC

Cada cop que es genera un nou tiquet, podem assumir que es genera un missatge d'avís a l'usuari enviat pel sistema.

Serà executat des dels següents procediments:

- **ticket_alta**

❖ **DW_sms_no_respostos_DEC**

Quan es genera un nou missatge, si l'emissor d'aquest és l'usuari que ha generat el tiquet i és el primer missatge per a aquell tiquet en el que l'emissor no és el sistema, s'ha de decrementar en 1 el nombre de missatges no respostos.

Serà executat des dels següents procediments:

- **missatge_alta**

6.3.11 DW_tickets_tancats

❖ **DW_tickets_tancats_INC**

Cada cop que es l'estat d'un tiquet passa a 'resolt' o 'cancel·lat' , s'ha d'incrementar en un la quantitat de tiquets tancats

Serà executat des dels següents procediments:

- **canviestat_alta** (quan l'estat passi a ser 'cancel·lat' o 'resolt')

❖ **DW_tickets_tancats_DEC**

En cas que s'elimini un tiquet, si aquest estava en estat 'resolt' o 'cancel·lat' s'ha de decrementar en un el nombre de tiquets tancats.

Serà executat des dels següents procediments:

- **ticket_baixa** (quan l'estat es 'cancel·lat' o 'resolt')

6.3.12 DW_tickets_per_canal

❖ **DW_tickets_per_canal_INC**

Cada cop que es genera un nou tiquet, s'ha d'incrementar el nombre de tiquets generats i concretament el nombre de tiquets generats pel canal donat en funció del país i el mes/any en què es genera.

Serà executat des dels següents procediments:

- **ticket_alta**

Paràmetres d'entrada:

- país on esta ubicada la seu on l'usuari genera el tiquet
- canal des de on es genera el ticket
- vector amb tots els canals existents al sistema

❖ DW_tickets_per_canal_DEC

En cas que s'elimini un tiquet o aquest passi a estar cancel·lat, s'hi ha de decrementar en un el nombre total de tiquets així com el nombre de tiquets oberts per aquell canal.

Serà executat des dels següents procediments:

- **ticket_baixa**
- **canviestat_alta**(quan l'estat passi a ser 'cancel·lat')

Paràmetres d'entrada:

- país on esta ubicada la seu on l'usuari genera el tiquet
- canal des de on es genera el tiquet

❖ DW_tickets_per_canal_MOD

Finalment, és possible que el canal d'entrada d'un tiquet es modifiqui, en aquest cas s'ha de prendre les mesures pertinents per a mantenir les dades correctes, és a dir, decrementar el nombre de tiquets per al canal anterior i incrementar el nombre de tiquets pel nou canal.

Serà executat des dels següents procediments:

- **ticket_modif_canal**

Paràmetres d'entrada:

- país on esta ubicada la seu on l'usuari genera el ticket
- canal des de on es va originar el ticket
- nou canal

6.4 Procediments de consulta

Finalment només resta generar els procediments de consulta a les taules del magatzem. En aquest cas com és obvi només es generarà un procediment per taula que farà una selecció d'aquelles dades que s'espera rebre (i alguna operació aritmètica si s'escau).

1. DW_temps_incidencies_prio_SEL

A l'hora de consultar aquesta taula, només haurem de fer una divisió del temps total entre la quantitat d'incidències per a cada prioritat donat un més/any.

2. DW_peticions_sobre_temps_SEL

Donat un mes/any i país, només caldrà fer la divisió entre el nombre de peticions resoltes per sobre del temps entre la quantitat total de peticions i multiplicar el resultat per 100 per a obtenir el percentatge.

3. DW_usuari_mes_incidencies_SEL

Caldrà retornar l'usuari amb la quantitat d'incidències més alt. Implica una ordenació descendent per la quantitat d'incidències i el retorn de la primera fila.

4. DW_top10_peticions_seu_SEL

Cal retornar els 10 serveis amb més quantitat de peticions per a la seu en qüestió. Implica una ordenació descendent per la quantitat de peticions de la seu i el retorn de les 10 primeres files.

5. DW_tickets_resolts_mes_SEL

Per a donar resposta només caldrà retornar el percentatge de tiquets resolts el mes actual respecte l'anterior. Per fer això haurem de demanar al sistema el mes i any actual, i calcular el mes anterior. Cal tenir en compte que si el més actual és 1 (gener) l'anterior ha de ser 12 i hem de restar per tant una unitat a l'any actual per obtenir-ne l'anterior. Un cop obtingudes aquestes dades només resta dividir els tiquets del mes actual entre els del mes anterior i multiplicar el resultat per 100 per obtenir el percentatge.

6. DW_servei_peticions_SEL

Caldrà retornar el servei amb la major quantitat de peticions associades. Això implica una ordenació descendent per la quantitat de peticions i el retorn de la primera fila.

7. DW_incidencies_seu_any_SEL

En aquest cas haurem d'ordenar en ordre descendent els registres per la quantitat d'incidències per l'any actual del sistema i retornar el primer valor pertinent a la seu amb més incidències per l'any actual.

8. DW_hores_tickets_L3_SEL

En aquest cas només cal fer una selecció de la quantitat d'hores per un any donat.

9. DW_incidencies_PM_SEL

Com en el cas anterior, només caldrà retornar la quantitat d'incidències per l'any actual del sistema.

10. DW_sms_no_respostos_SEL

Per a l'any actual del sistema, haurem de dividir la quantitat de tiquets no respostos entre el total de tickets per a aquell any i multiplicar el resultat per 100 per obtenir el percentatge de sms no respostos per l'any en curs.

11. DW_tickets_tancats_SEL

Aquesta taula només conté una tupla amb un sol atribut que és el que s'ha de retornar.

12. DW_tickets_per_canal_SEL

Donat un més/any i país només caldrà, per a cada canal fer el percentatge dels tiquets d'aquell canal respecte al total. Això és dividir la quantitat de tiquets per aquell canal entre el total de tiquets (sempre en funció del mes/any i país) i multiplicar el resultat per 100 per obtenir-ne el percentatge.

7. Proves

En el darrer apartat de desenvolupament de la nostra base de dades, el que farem serà crear totes les dades i crides necessàries per a poder comprovar el correcte funcionament de la base de dades i el magatzem de dades en totes les seves variants: inserció de dades, correcte funcionament dels registres, sortides esperades, correcta gestió d'excepcions, temps de resposta sol·licitats i totes les qüestions referents a la correctesa del producte que hem generat.

Per a poder fer totes les proves pertinents, el primer que haurem de generar és un volum de dades inicials que pobli la nostra base de dades. Un cop poblada d'un mínim de dades, el següent pas serà fer una anàlisi d'aquells casos i funcionalitats que volem gestionar en cada cas, per poder fer un seguit d'insercions i modificacions de dades i comprovar-ne la reacció de la base de dades, assegurant-nos que aquesta és la que esperem en cadascun dels casos. Finalment mostrarem alguns dels resultats dels jocs de proves que demostrin el correcte funcionament del sistema.

7.1 Volum de dades inicial

Tal com ja hem comentat anteriorment, el primer que farem serà poblar la base de dades amb un mínim de dades que després ens ajudin a veure la correctesa del sistema a través de jocs de proves.

En el meu plantejament d'aquest capítol, les dades que introduïrem a la base de dades en aquest punt són correctes en el sentit més estricte de la paraula: Són dades que la base de dades espera rebre, sense cap problema en forma ni referencia a altres dades (claus foranes). Per tant, la introducció d'aquestes dades inicials no farà saltar cap excepció a la nostra base de dades. Les dades introduïdes a través dels procediments d'alta/baixa/modificació especificats a l'apartat [Procediments](#) es poden trobar a l'[Annex](#).

Recordem que aquestes funcions sempre tenen un paràmetre de sortida anomenat RSP que podem recollir per veure el resultat de l'execució o bé podem consultar la taula Log que és l'encarregada de desar les crides a les funcions. Si ho fem, podem veure que efectivament totes les dades s'han introduït a la base de dades de manera correcta.

Ara que la base de dades està mínimament poblada, i podem veure que en casos ideals les dades s'introdueixen correctament, és moment de determinar els casos que volem forçar i examinar per a veure'n el correcte funcionament.

7.2 Anàlisi dels casos i funcionalitats

En aquest apartat el que es farà serà examinar, per a cada entitat, aquells casos que es volen forçar a la base de dades per a poder determinar el correcte funcionament dels procediments que hem generat. Això és, per exemple, introduir dades incorrectes, que violin restriccions de la base de dades i/o que generin aquelles excepcions que hem tractat, així com dades correctes, tant per les altes, les baixes i les modificacions de les diferents entitats que conformen la nostra base de dades).

Per una altra banda, per a comprovar el correcte funcionament del magatzem, el que farem serà generar aquells casos en els quals les taules del magatzem han de ser actualitzades per a poder comprovar si realment tots els casos han sigut coberts i funcionen com s'espera.

Sent així, anem a examinar tots els casos que volem generar i posteriorment crearem els jocs de proves pertinents per a poder cobrir-los. Els casos cobriran les següents situacions per a cada taula:

- Alta correcta
- Alta incorrecta (per diferents motius)
- Baixa incorrecta (per diferents motius)
- Modificació incorrecta (per diferents motius)
- Modificació correcta
- Baixa correcta

Separarem aquests casos per entitats de la base de dades per tal de poder-ho presentar de manera ordenada. A més, els separarem al seu torn en 2 gran blocs: base de dades operativa i magatzem de dades.

Recordem que en aquest punt només veurem els casos i les funcionalitats a estudiar, mentre que el resultat de l'execució dels jocs la podem trobar a l'[Annex](#).

7.2.1 Base de dades operativa

- País
 - ☒ Alta d'un país inexistent
 - ☒ Alta d'un país que ja existeix
 - ☒ Baixa d'un país on hi ha ubicada una seu
 - ☒ Baixa d'un país
- Seu
 - ☒ Alta d'una seu
 - ☒ Alta d'una seu que ja existeix
 - ☒ Alta d'una seu amb algun valor nul no permès
 - ☒ Alta d'una seu a un país que no existeix
 - ☒ Baixa d'una seu on hi ha inscrit algun usuari
 - ☒ Baixa d'una seu

- Usuari
 - ☒ Alta d'un usuari nou
 - ☒ Alta d'un usuari que ja existeix
 - ☒ Alta d'un usuari amb algun valor nul no permès
 - ☒ Alta d'un usuari a una seu que no existeix
 - ☒ Modificació de la seu d'un usuari que es nul·la
 - ☒ Modificació de la seu que no existeix
 - ☒ Modificació de la seu
 - ☒ Baixa d'un usuari

- NivellSuport
 - ☒ Alta d'un nivell de suport
 - ☒ Alta d'un nivell que ja existeix
 - ☒ Baixa d'un nivell al que correspon algun grup de suport
 - ☒ Baixa d'un nivell de suport

- GrupSuport
 - ☒ Alta d'un grup de suport
 - ☒ Alta d'un grup que ja existeix
 - ☒ Alta d'un grup amb algun valor nul no permès
 - ☒ Alta d'un grup amb un nivell no existent
 - ☒ Baixa d'un grup relacionat amb algun tiquet
 - ☒ Modificació del nom amb nom nul
 - ☒ Modificació del nom
 - ☒ Baixa d'un grup de suport

- Tècnic
 - ☒ Alta d'un tècnic
 - ☒ Alta d'un tècnic que ja existeix
 - ☒ Alta d'un tècnic amb un valor nul no permès
 - ☒ Alta d'un tècnic que pertany a un grup de suport no existent
 - ☒ Baixa d'un tècnic que està relacionat amb algun canvi d'estat
 - ☒ Modificació del grup a un grup no existent
 - ☒ Modificació del grup de suport
 - ☒ Baixa d'un tècnic

- Prioritat
 - ☒ Alta d'una prioritat
 - ☒ Alta d'una prioritat que ja existeix
 - ☒ Baixa d'una prioritat a la que pertany algun ticket o servei
 - ☒ Baixa d'una prioritat

- GrupServei
 - ☒ Alta d'un grup de servei
 - ☒ Alta d'un grup que ja existeix
 - ☒ Baixa d'un grup de servei

- Servei
 - ☒ Alta d'un servei
 - ☒ Alta d'un servei que ja existeix
 - ☒ Alta d'un servei amb algun valor nul no permès
 - ☒ Baixa d'un servei sol·licitat per algun tiquet o ofert per algun grup
 - ☒ Modificació del grup del servei a un grup no existent
 - ☒ Modificació del temps mig a un temps mig nul
 - ☒ Modificació de la prioritat a una prioritat no existent
 - ☒ Modificació de la descripció
 - ☒ Modificació del grup
 - ☒ Modificació de la prioritat
 - ☒ Modificació del temps mig
 - ☒ Baixa d'un servei

- OfertaGrupServei
 - ☒ Alta d'una oferta
 - ☒ Alta d'una oferta ja existent
 - ☒ Alta d'una oferta amb algun valor nul no permès
 - ☒ Baixa d'una oferta

- TipusTicket
 - ☒ Alta d'un tipus
 - ☒ Alta d'un tipus que ja existeix
 - ☒ Baixa d'un tipus al que pertany algun tiquet
 - ☒ Baixa d'un tipus

- Canal
 - ☒ Alta d'un canal
 - ☒ Alta d'un canal que ja existeix
 - ☒ Baixa d'un canal al que pertany algun tiquet
 - ☒ Baixa d'un canal

- CrecioTicket
 - ☒ Alta d'un mètode de creació
 - ☒ Alta d'un mètode de creació que ja existeix
 - ☒ Baixa d'un mètode de creació al que pertany algun tiquet
 - ☒ Baixa d'un mètode de creació

- Ticket
 - ☒ Alta d'un tiquet
 - ☒ Alta d'un tiquet que ja existeix
 - ☒ Alta d'un tiquet amb algun valor nul no permès
 - ☒ Alta d'un tiquet amb un servei no existent
 - ☒ Alta d'un tiquet amb un prioritat no existent
 - ☒ Alta d'un tiquet amb un tipus no existent
 - ☒ Alta d'un tiquet amb un canal no existent
 - ☒ Alta d'un tiquet amb un mètode de creació no existent
 - ☒ Alta d'un tiquet amb un usuari no existent
 - ☒ Baixa d'un tiquet amb una assignació vigent
 - ☒ Modificació d'un canal no existent

- ☒ Modificació d'un tipus no existent
- ☒ Modificació del canal
- ☒ Modificació del mètode de creació
- ☒ Baixa d'un tiquet

- AssigTicketGrup
 - ☒ Alta d'una assignació
 - ☒ Alta d'una assignació que ja existeix
 - ☒ Alta d'una assignació on el nivell del grup és inferior al nivell del grup de la assignació vigent
 - ☒ Alta d'una assignació a un grup que no ofereix el servei que el tiquet sol·licita
 - ☒ Baixa d'una assignació

- EstatTicket
 - ☒ Alta d'un estat
 - ☒ Alta d'un estat que ja existeix
 - ☒ Baixa d'un estat al que pertany algun canvi d'estat d'un tiquet
 - ☒ Baixa d'un estat

- CanviEstat
 - ☒ Alta d'un canvi d'estat
 - ☒ Alta d'un canvi d'estat efectuat per un tècnic que no pertany al grup associat actualment amb el tiquet en qüestió

- EstatSms
 - ☒ Alta d'un estat
 - ☒ Alta d'un estat existent
 - ☒ Baixa d'un estat al que pertany algun missatge
 - ☒ Baixa d'un estat

- Missatge
 - ☒ Alta d'un missatge
 - ☒ Alta d'un missatge existent
 - ☒ Alta d'un missatge amb algun valor nul no permès
 - ☒ Modificació d'un estat que no existeix
 - ☒ Modificació de l'estat

7.2.2 Data Warehouse

La generació dels jocs de proves per a les taules del magatzem és una mica més farragosa que la de la base de dades operativa, ja que en el cas anterior, podem executar un seguit d'instruccions en bloc i després comprovar la sortida de la variable RSP a la taula log per veure si el procediment s'ha comportat com esperàvem en funció de les variables d'entrada donades. En aquest cas, haurem de fer la consulta a taula, fer els canvis pertinents i tornar a consultar la taula per veure que realment aquesta s'ha modificat tal com s'espera. Per tant en aquest procés és testejant tant els procediments de consulta que són els que cridarem nosaltres de manera manual, com tots els procediments que s'encarreguen de modificar les taules del magatzem en funció de les accions que generem sobre la base de dades. Per a fer-ho dividirem aquestes proves per a cada taula del magatzem. Anem doncs a examinar els casos que provocarem per a veure'n els resultats.

➤ DW_temps_incidencies_prio

- **Estat inicial:** al sistema existeixen tiquets de tipus incidència de dos prioritats diferents: 1 i 2. I cap tiquet està en estat de resol, per tant el temps mig de resolució es 0 (inexistent).
- **Canvis a la base de dades:**
 - ☑ Canvi d'estat d'incidència de prioritat 1 a resolta
 - ☑ Canvi d'estat d'incidència de prioritat 2 a resolta
 - ☑ Creació de nova incidència de prioritat 1 a estat 'Assignat'
 - ☑ Canvi d'estat de la nova incidència de prioritat 1 a 'Resolta'
 - ☑ Creació de nova incidència de prioritat 4 a estat 'Assignat'
 - ☑ Canvi d'estat de la nova incidència de prioritat 4 a 'Resolta'

➤ DW_peticions_sobre_temps

- **Estat inicial:** al sistema existeixen tiquets de tipus petició de dos usuaris, un ubicat a Itàlia i un altre a Espanya. No obstant cap dels dos tiquets està resolt de manera que la consulta no retorna res.
- **Canvis a la base de dades:**
 - ☑ Canvi d'estat de petició de l'usuari ubicat a Espanya a resolta
 - ☑ Creació d'una altra petició a per un usuari ubicat a la seu d'Espanya
 - ☑ Canvi d'estat d'aquesta petició de l'usuari ubicat a Espanya a resolta (per sota del temps mig de resolució perquè l'acabem de crear).

➤ DW_usuari_mes_incidencies

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que l'usuari amb més incidències al sistema es l'usuari sandra@corp.com amb 3 incidències. Per tant i per veure que realment funciona crearem 6 incidències més generades per un sol usuari.
- **Canvis a la base de dades:**
 - ☑ Creació de 6 incidències per un usuari

➤ DW_top10_peticions_seu

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que la seu S001 ha sol·licitat 2 serveis diferents (SERV003 i SERV002) un total de una vegada cadascun d'ells. Per una altre banda la seu S003 ha sol·licitat un sol cop el servei SERV005. Anem a crear un seguit d'incidències de diferents usuaris per veure que el que generem es correspon amb el contingut de la taula i amb la sortida de la consulta.
- **Canvis a la base de dades:**
 - ☑ Creació de 4 peticions al servei SERV004 de la seu S001
 - ☑ Creació de 5 peticions al servei SERV001 de la seu S002
 - ☑ Creació de 3 peticions al servei SERV003 de la seu S003
 - ☑ Creació de 4 peticions al servei SERV005 de la seu S001
 - ☑ Creació de 1 peticions al servei SERV003 de la seu S001

➤ DW_tickets_resolts_mes

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen, pel més actual, 6 peticions resoltes. Atès que pel mes anterior a l'actual no existeix cap entrada, si executem el procediment no ens retornarà cap resultat. En aquest cas haurem de forçar la introducció de dades pel mes anterior d'una manera no natural per tal de poder fer les comprovacions necessàries.
- **Canvis a la base de dades:**
 - ☑ Creació d'una entrada forçada pel més anterior amb 3 tickets resolts
 - ☑ Creació d'una entrada forçada pel més anterior amb 18 tickets resolts

➤ DW_servei_peticions

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen entrades per a 5 serveis, SERV001, SERV003 i SERV005 sol·licitats 5 vegades, SERV004 sol·licitats 4 vegades i SERV002 sol·licitat 1 vegada.
- **Canvis a la base de dades:**
 - ☑ Creació d'una petició que sol·liciti el servei SERV005
 - ☑ Cancel·lació de la petició anterior
 - ☑ Creació de 2 peticions que sol·licitin el servei SERV004

➤ DW_incidencies_seu_any

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen 9 incidències per la seu S001 i 1 per la seu S002, ambdós per l'any 2018. Per tant si fem la consulta, atès que l'any en curs es el 2018, ens retornarà la seu S001.
- **Canvis a la base de dades:**
 - ☑ Creació de 12 consultes per la seu S003

➤ DW_hores_tickets_L3

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen no existeix cap entrada ja que no hi ha cap tiquet assignat a un grup de suport L3. Per a generar els jocs de proves haurem de forçar que el sistema esperi entre l'assignació del tiquet a un grup L3 i posteriorment la reassignació del mateix tiquet a un grup de problem management (moment en el qual es comptabilitza els nombre d'hores que el tiquet ha estat assignat a un grup de suport L3. També es passarà a resolt un tiquet assignat a un grup de L3.
- **Canvis a la base de dades:**
 - ☑ Creció d'un tiquet i assignació a un grup de suport de nivell L3
 - ☑ Assignació del tiquet anterior al grup de suport PM
 - ☑ Creació d'un tiquet i assignació a un grup de suport de nivell L3
 - ☑ Canvi d'estat del tiquet anterior a resolt

➤ DW_incidencies_PM

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeix 1 incidència assignada a un grup de nivell PM. Per tant el que farem serà crear algunes incidències assignades a grups de PM i després donar de baixa les assignacions. Cal veure que la base de dades controla que no es pugui fer un downgrade de grup (és a dir que si hi ha una assignació a un grup L3, no es pot fer una assignació posterior pel mateix tiquet a un grup de nivell inferior).
- **Canvis a la base de dades:**
 - ☑ Creció de dos incidències assignades a grups PM
 - ☑ Baixa d'una de les incidències anteriors

➤ DW_sms_no_respostos

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen 59 missatges enviats i 58 no respostos. Això és perquè amb el consultor es va arribar a l'enteniment que es podia considerar que cada cop que es genera un nou tiquet, el sistema envia un missatge a l'usuari que l'ha generat (serà responsabilitat del programador de la aplicació encarregar-se de l'enviament del missatge). Per tant al llarg dels jocs de proves i el volum de dades inicials hem generat 59 missatges, dels quals només 1 ha estat respostos (al propi volum inicial de dades).El que farem per testejar aquesta funcionalitat és:
- **Canvis a la base de dades:**
 - ☑ Respondre 10 missatges de tickets existents

➤ DW_tickets_tancats

- **Estat inicial:** En aquest moment, si visitem la taula podem veure que existeixen 8 tiquets tancats, això és que 8 tiquets han estat resolts o cancel·lats per l'usuari que els ha creat. Per a poder veure el correcte funcionament de la base de dades generarem els següents casos
- **Canvis a la base de dades:**
 - ☑ Canvi d'estat de 3 tiquets existents a Resolt
 - ☑ Cancel·lació de 2 tiquets existents

➤ DW_tickets_per_canal

- **Estat inicial:** En aquest moment, si visitem la taula podem veure la següent entrada

PAIS	MES	ANI	CANAL	QTT_TICKETS_CANAL	QTT_TICKETS_TOTAL
Espanya	5	2018	Web	1	32
Espanya	5	2018	Xat	31	32
Espanya	5	2018	Telefon	0	32
Italia	5	2018	Web	0	14
Italia	5	2018	Xat	13	14
Italia	5	2018	Telefon	1	14

Per tant, les sortides a les següents comandes son les que podem observar:

```
DW_tickets_per_canal_SEL(05, 2018, 'Espanya');  
Telefon : 0  
Web : 3,125  
Xat : 96,875
```

```
DW_tickets_per_canal_SEL(05, 2018, 'Italia');  
Telefon : 7,1428  
Web : 0  
Xat : 92,8571428
```

- **Canvis a la base de dades:**
 - ☑ Creació de 5 tiquets per Web a Espanya
 - ☑ Creació de 2 tiquets per Telefon a Italia
 - ☑ Creació d'un usuari a una seu de França i creació d'un tiquet per Web

8. Conclusions

8.1 Conclusions del treball

Al llarg del desenvolupament del treball he anat fixant conclusions en diferents aspectes, tant tècnics del mateix treball, com de l'execució del projecte, planificació temporal, gestió personal, i un llarg etcètera. Així que per a exposar-ho d'una manera ordenada, presentaré un seguit de punts a tall de conclusió.

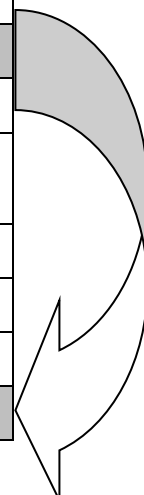
- **La importància de la planificació temporal:** La meva experiència personal vers al desenvolupament de projectes informàtics és pràcticament nul·la atès que no m'he introduït encara en aquest sector laboral. Això fa que en aquesta ocasió hagi pogut veure amb total claredat la importància de la planificació temporal en el desenvolupament de projectes, tal com ja se'ns ha anat dient a vàries de les assignatures del grau. Una bona planificació temporal et dóna una visió del ritme de treball que has de portar i marca els objectius a assolir al llarg del temps, fet essencial per a poder ubicar-te dins l'espai de temps comprés entre el qual portes desenvolupat i allò que queda per fer. Una mala planificació temporal, per contra, pot ser nefast si no es detecta a temps, impossibilitant a tots els efectes no només la finalització del projecte sinó una correcta presa de decisions a tots els nivells donada la precipitació que la manca de temps porta amb si.
- **El benefici personal en l'assoliment d'objectius:** Sorprenentment, una de les coses que més m'ha motivat durant l'execució del projecte ha sigut l'assoliment dels objectius marcats al llarg del temps a la planificació prèvia. L'assoliment d'objectius m'ha reportat satisfacció que es traduïa en concentració i ganes de seguir endavant amb el projecte amb les millors praxis possibles de desenvolupament tant del mateix projecte com del producte que aquest acabaria generant, la base de dades.
- **La importància de la metodologia:** Tant pel que respecta a les fases del projecte, com al desenvolupament de la base de dades, he pogut recalcar la importància que té el fet de seguir una metodologia estricta a l'hora de generar la documentació o els productes del projecte. Aquesta metodologia, apresada a assignatures del grau com les de base de dades o gestió de projectes han sigut claus en el correcte desenvolupament del treball.
- **El desconeixement i els errors:** La font més gran d'errors que he pogut detectar en el desenvolupament del projecte ha sigut el desconeixement. Aquest fet va ser el que va reportar la majoria de problemes a l'hora de generar el codi de la base de dades. Això no obstant, gràcies al professor consultor per una banda, i a les eines que he pogut obtenir a les assignatures del grau, així com a les competències transversals apresades al llarg de la carrera per una altra, he pogut cercar la informació en fonts fiables, integrar-la als coneixements que ja tinc, aplicar-la al meu cas i solvatar tots els problemes que han anat sorgint.

- **L'aprenentatge a través de l'execució:** Si bé és cert que això queda de manifest des de les primeres assignatures de la carrera, he pogut reafirmar el fet de la importància de l'aprenentatge a través de l'execució. I és que encara que estigui clara la teoria, és en el desenvolupament i execució d'aquesta teoria prèviament assimilada en el que resideixen els problemes, els grans petits detalls i les particularitats de cada situació que t'obliguen a cercar, refrescar o completar el coneixement que ja portem per aplicar-lo en casos pràctics. Això produeix, si més no en el meu cas, que els conceptes quedin fixats al meu cap d'una manera molt més clara i permanent.
- **La importància de les consultes:** Finalment volia deixar latent la importància de poder consultar qualsevol dubte i ser respost en un període de temps molt curt. Pel que respecta aquest fet, el professor consultor ha sigut clau per a la solució de problemes o dubtes referents a l'execució del projecte.

8.2 Anàlisi sobre l'assoliment d'objectius

En aquest punt analitzarem l'assoliment dels objectius al llarg del temps (feina entregada a les PAC respecte a la feina que vam planificar poder lliurar a cadascuna d'elles) i analitzarem de manera crítica en el conjunt de treball quins objectius inicials s'han pogut aconseguir. Així doncs, en primera instància veurem una taula que representa aquella feina que vam planificar lliurar a cadascuna de les entregues respecte a aquella feina que finalment va poder ser lliurada:

Concepte	Data	Tasca planificada	Lliurament
PAC 1	05/03	Lectura de l'enunciat i documentació	Sí
		Pla de treball	Sí
PAC 2	09/04	Tria del programari	Sí
		Instal·lació i configuració	Sí
		Tria del conjunt de dades	Sí
		Generació del diagrama UML	Sí
		Restriccions textuais	Sí
		Creació de les taules	Sí
		Generació del volum de dades	Reubicat(*)
PAC 3	07/05	Creació de procediments	Sí
		DW: Anàlisi de dades i estructura de taules	Sí
		DW: Creació de les taules	Sí
		DW: Procediments d'actualització	Sí
		DW: Procediments de consulta	Sí
		Generació del volum de dades(*)	Sí



		Creació dels jocs de proves	Sí
		Proves a la base de dades	Sí
PAC 4	11/06	Síntesi i generació de diapositives	Sí
		Creació del guió	Sí
		Preparació de l'entorn	Sí
		Preparació de la presentació	Sí
		Gravació i muntatge	Sí
		Generació de l'informe de CT	Sí

(*) **Generació del volum de dades:** En una primera instància es va considerar aquesta tasca per a ser executada després de la creació de les taules i prèvia a la segona entrega parcial del treball. Això no obstant, cal reconsiderar-ho atès que en aquest punt encara no tenim fets els procediments d'inserció de dades a i per tant per a poblar la base de dades en aquest moment, caldria fer-ho de manera directa fent ús de sentències "INSERT INTO".

Cal tenir en compte però que les funcions d'alta/baixa/modificació són les que contindran tots els controls que s'han de fer sobre les dades, tant les restriccions textuais com la tasca de nodrir les futures taules que compondran el magatzem. Per tant, en aquest punt no és una tasca útil ni aporta res al treball poblar la base de dades amb sentències INSERT INTO.

És per això que aquesta tasca es reubica al treball per fer-se després de la creació dels procediments, de les taules que componen el magatzem i els procediments que el poblen i previ als jocs de proves.

Tret d'aquest punt tal com es pot observar a la taula s'ha aconseguit lliurar a cada PAC aquelles tasques planificades amb anterioritat, de manera que en el seu global podem considerar la planificació de tasques com exitosa.

No obstant això, cal ser crítics en l'aspecte en què és possible que encara que les tasques hagin estat completades a cada lliurable, que era l'objectiu, el balanceig entre aquestes possiblement no era el més adequat, atès que les tasques compreses entre el 09/04 i el 07/05 (és a dir aquelles que s'esperava tenir per a la PAC3 i que així es va fer) portaven una càrrega de feina molt superior a la resta d'entregues, sent aquestes tasques en la meua opinió el 70% del còmput total de feina a fer en el transcurs del projecte, ja que inclou la creació de tots els procediments per a gestionar la base de dades, la creació de tot el magatzem (taules, procediments d'actualització i de consulta) i la generació de tots els jocs de proves.

8.3 Anàlisi del seguiment de la planificació

Anem ara a veure una comparativa entre les tasques previstes amb el seu cost temporal respecte a les tasques resultants de l'elaboració del projecte amb el cost temporal que realment han tingut. Abans però, cal ser crític en alguns aspectes clarament presents en la planificació inicial del projecte:

El primer és l'aparició de noves tasques que no van ser planificades prèviament. Concretament parlem del bloc de disseny lògic de la base de dades, que si bé és una traducció intermèdia entre el model conceptual en UML i les sentències SQL del disseny físic, s'ha d'efectuar per a seguir un correcte disseny de la base de dades. A més aquest bloc també aglomera les tasques de la identificació de possibles paranyes de disseny presents al treball, així com la normalització de les relacions. Més concretament:

- **Traducció del model conceptual al model lògic:** Un cop ens assegurem que el nostre model conceptual no conté cap parany de disseny, passarem a la seva traducció al model lògic tenint en compte que la nostra base de dades serà una base de dades relacional.
- **Normalització:** Un cop obtenim l'esquema lògic de la base de dades és el moment, si s'escau, de normalitzar aquelles anomalies presents en les relacions, causades per les dependències existents entre atributs d'aquesta^[11].

El segon concepte és l'encavalcament existent entre tasques que no es va tenir en compte a la planificació inicial. Més concretament entre el disseny de la base de dades i el del magatzem. Això és per un desconeixement personal sobre el funcionament i estructura d'un magatzem de dades, atès que mai havia treballat ni vist cap. No obstant amb el projecte he pogut descobrir el seu funcionament i en el transcurs d'aquest descobriment he vist que en el moment del disseny de la base de dades, inevitablement s'ha de tenir present de quina manera es desaran les dades al magatzem i sobretot quines accions sobre la base de dades faran que aquestes dades del magatzem s'hagi de modificar/actualitzar. Amb tot això, existeix un encavalcament dels blocs de disseny de la base de dades amb els del magatzem de dades que no estan reflectits a la planificació inicial.

Finalment, val a dir que en la majoria dels blocs, la planificació va ser bastant acurada. A ressaltar l'excés d'hores planificades per a l'elaboració de la memòria per una banda, i la mancança d'hores planificades pels jocs de proves de la base de dades. Tret d'aquests dos blocs, la resta d'hores invertides entren dins la forquilla planificada en la majoria dels casos, o la desviació és mínima.

Dit això, aquí presentem la comparativa d'hores.

[11] *Disseny lògic de bases de dades: PID_00220510*. Xavier Bargués Illa. Universitat Oberta de Catalunya.

Tasca	Hores planificació	Hores invertides
Feina inicial	16-20	18
Lectura de l'enunciat i documentació	4-6	4
Pla de treball	12-14	14
Preparació de l'entorn	4-8	5
Tria de programari	2-4	3
Instal·lació i configuració	2-4	2
Disseny conceptual	34-40	37
Tria del conjunt de dades	10	5
Generació del diagrama UML	20-24	30
Restriccions textuais	4-6	2
Disseny lògic	-	6
Traducció a model lògic	-	2
Normalització	-	4
Disseny Físic	38-42	28
Creació de taules	10-12	8
Creació de procediments	28-30	20
Data Warehouse	22-28	30
Estudi de dades i estructura de taules	10	6
Creació de les taules	2-4	4
Procediments d'actualització	10	16
Procediments de consulta	4	4
Proves	10-12	30
Generació de dades	4-6	8
Creació dels jocs	6	12
Proves a la base de dades	4-6	10
Presentació en vídeo	30-36	32
Síntesi i generació de diapositives	8-10	10
Creació del guió	4-6	8
Preparació de l'entorn	4	2
Preparació de la presentació	4	2
Gravació i muntatge	10-12	10
Avaluació de les CT	2-4	2
Generació de l'informe	2-4	2
Documentació a lliurar	158-164	120
Pla de treball	12-14	14
Memòria	120	80
Base de dades	18-20	20
Jocs de proves	8-10	6

8.4 Línies de treball futur

Pel que respecta al futur referent al producte generat per aquest projecte, podem trobar diversos punts que aquest projecte no a cobert bé perquè no es demanava o bé perquè a tots els efectes, queda fora de l'abast d'aquest. Algunes de les línies de treball futur són:

- **Desenvolupament de l'aplicació:** Una aplicació que faci ús d'aquesta base de dades permetent als usuaris generar tiquets, efectuar interaccions sobre el sistema, enviar missatges i tot el que fa referència a la generació de consultes i peticions per part dels usuaris finals del sistema.
- **BI:** En aquest moment, la base de dades compta amb un magatzem que aporta informació que es pot fer servir per a generar informes que puguin ajudar a la presa de decisions estratègiques per part de la direcció de l'empresa.
- **Augment del magatzem:** Pel que respecta al magatzem, cal recalcar que aquest aporta les dades requerides per les consultes sol·licitades al projecte Això no obstant, amb el temps la quantitat de consultes que es desitja fer al magatzem pot augmentar, i en conseqüència ho haurien de fer tant les estructures que conformen el magatzem com els procediments que en gestionen les dades.
- **Unificació de les bases de dades:** Cal recalcar que aquesta base de dades conforma una petita porció de les dades amb les quals l'empresa compta. Concretament només des a i gestiona aquelles dades referents a la resolució dels tiquets generats pels usuaris. En un futur, es podria considerar la integració total d'aquestes dades a la base de dades global ja existent, si això pogués reportar algun benefici per l'empresa o de caràcter pràctic en la gestió de les dades o l'elaboració d'informació que se'n desprèn.
- **Augment de les dades:** Tal com ja s'ha anat explicant al desenvolupament de la base de dades, aquesta ha estat pensada per a facilitar la introducció de noves dades generant taules que contenen enumeracions de possibles atributs tals com els tipus de tiquets que l'usuari pot generar, els mètodes de creació, els canals pels quals es notifiquen i els estats que pot prendre, entre d'altres.

9. Glossari

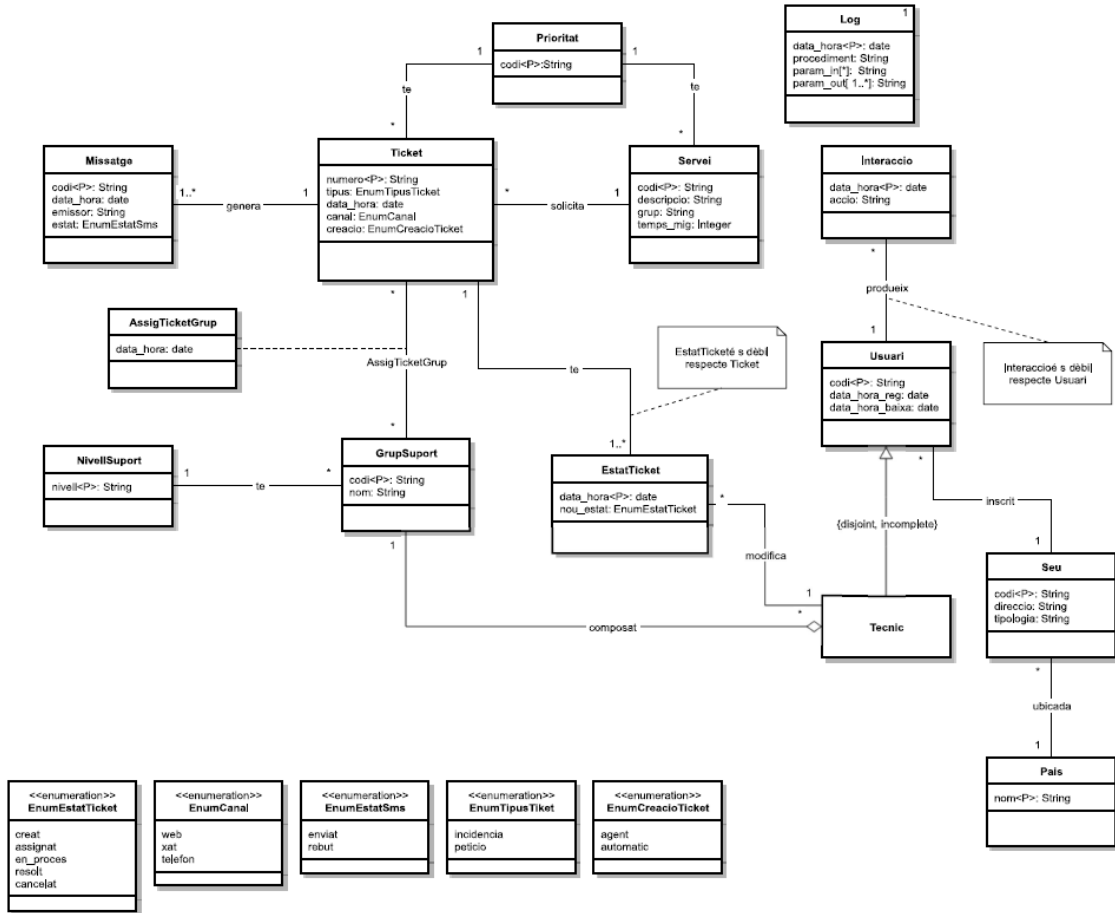
- **Entitat:** Representa un objecte o concepte de la realitat que es diferencia únicament la resta
- **Atribut:** Característiques que defineixen i/o identifiquen una entitat
- **Relació:** Concepte existent entre dos entitats que les uneix. També es pot referir en l'àmbit lògic de base de dades a la representació d'una entitat.
- **Magatzem de dades / Data warehouse:** Col·lecció de dades variables en el temps que són d'utilitat per a la presa de decisions, generació d'informes...
- **ER:** Entitat Relació (Model): Model de dades que permet representar entitats rellevants per a un sistema d'informació, els seus atributs i les relacions existents entre elles.
- **UML:** Unified Modeling Language: Llenguatge gràfic per a visualitzar, especificar, documentar i construir un sistema.
- **DB/BD:** Data Base / Base de Dades: Conjunt de dades pertinents a un mateix context i emmagatzemades sistemàticament per al seu ús.
- **SGBD:** Sistema Gestor de Base de Dades: Conjunt de programes que permeten l'emmagatzematge, modificació i extracció de la informació continguda en una base de dades
- **1FN:** Primera forma normal de la teoria de la normalització.
- **2FN:** Segona forma normal de la teoria de la normalització.
- **3FN:** Segona forma normal de la teoria de la normalització.
- **FNBC:** Forma normal de Boyce-Codd de la teoria de la normalització.

10. Bibliografia

- **Model ER:** https://es.wikipedia.org/wiki/Modelo_entidad-relaci%C3%B3n [28/03/2018]
- **Llenguatge UML:** https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado [28/03/2018]
- **Base de dades:** https://es.wikipedia.org/wiki/Base_de_datos [28/03/2018]
- **Sistema Gestor de Base de Dades:** https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos [28/03/2018]
- **Disseny conceptual de bases de dades: PID_00220512.** Jordi Casas Roma. Universitat Oberta de Catalunya. [28/03/2018]
- **Disseny lògic de bases de dades: PID_00220510.** Xavier Bargués Illa. Universitat Oberta de Catalunya. [28/03/2018]
- **Disseny lògic de bases de dades: PID_00220510.** Xavier Bargués Illa. Universitat Oberta de Catalunya. [28/03/2018]
- **Primera forma normal:** https://es.wikipedia.org/wiki/Primera_forma_normal [04/04/2018]
- **Segona forma normal:** https://es.wikipedia.org/wiki/Segunda_forma_normal [04/04/2018]
- **Tercera forma normal:** https://es.wikipedia.org/wiki/Tercera_forma_normal [04/04/2018]
- **Forma normal de Boyce-Codd:** https://es.wikipedia.org/wiki/Forma_normal_de_Boyce-Codd [04/04/2018]
- **Oracle Database Online Documentation:** <https://docs.oracle.com/en/database/> [05/04/2018]
- **W3Schools:** <https://www.w3schools.com/> [05/04/2018]
- **Oracle Error Messages:** http://www.oracle.com/pls/db92/db92.error_search?prefill=ORA- [11/04/2018]

11. Annexos

11.1 Versió prèvia del mapa conceptual



V1

11.2 Taules de la base de dades operativa

```
CREATE TABLE Pais (  
    nom VARCHAR2 (50 CHAR) CONSTRAINT NN_Pais_nom NOT NULL,  
    CONSTRAINT PK_Pais PRIMARY KEY (nom)  
);
```

```
CREATE TABLE Seu (  
    codi VARCHAR2 (20 CHAR) CONSTRAINT NN_Se_u_codi NOT NULL,  
    direccio VARCHAR2 (50 CHAR) CONSTRAINT NN_Se_u_direccio NOT NULL,  
    tipologia VARCHAR2 (50 CHAR),  
    pais VARCHAR2 (50 CHAR) CONSTRAINT NN_Se_u_pais NOT NULL,  
    CONSTRAINT PK_Se_u PRIMARY KEY (codi),  
    CONSTRAINT FK_Se_u_Pais FOREIGN KEY (pais) REFERENCES Pais (nom)  
);
```

```
CREATE TABLE Usuari (  
    mail VARCHAR2 (50 CHAR) CONSTRAINT NN_Usuari_mail NOT NULL,  
    data_hora_reg TIMESTAMP CONSTRAINT NN_Usuari_data_hora_reg NOT NULL,  
    data_hora_baixa TIMESTAMP,  
    seu VARCHAR2 (20 CHAR) CONSTRAINT NN_Usuari_seu NOT NULL,  
    CONSTRAINT PK_Usuari PRIMARY KEY (mail),  
    CONSTRAINT FK_Usuari_Se_u FOREIGN KEY (seu) REFERENCES Seu (codi),  
    CONSTRAINT CH_data_reg_baixa CHECK (data_hora_baixa IS NULL  
    OR data_hora_baixa > data_hora_reg) /*RT4*/  
);
```

```
CREATE TABLE Interaccio (  
    usuari VARCHAR2 (50 CHAR) CONSTRAINT NN_Interaccio_usuari NOT NULL,  
    data_hora TIMESTAMP CONSTRAINT NN_Interaccio_data_hora NOT NULL,  
    accio VARCHAR2 (50 CHAR) CONSTRAINT NN_Interaccio_accio NOT NULL,  
    CONSTRAINT PK_Interaccio PRIMARY KEY (usuari, data_hora),  
    CONSTRAINT FK_Interaccio_Usuari FOREIGN KEY (usuari) REFERENCES Usuari (mail)  
);
```

```
CREATE TABLE NivellSuport (  
    nivell VARCHAR2 (20 CHAR) CONSTRAINT NN_NivellSuport_nivell NOT NULL,  
    valor INTEGER CONSTRAINT NN_NivellSuport_valor NOT NULL,  
    CONSTRAINT PK_NivellSuport PRIMARY KEY (nivell)  
);
```

```
CREATE TABLE GrupSuport (  
    codi VARCHAR2 (20 CHAR) CONSTRAINT NN_GrupSuport_codi NOT NULL,  
    nom VARCHAR2 (50 CHAR) CONSTRAINT NN_GrupSuport_nom NOT NULL,  
    nivell VARCHAR2 (20 CHAR) CONSTRAINT NN_GrupSuport_nivell NOT NULL,  
    CONSTRAINT PK_GrupSuport PRIMARY KEY (codi),  
    CONSTRAINT FK_GrupSuport_nivell FOREIGN KEY (nivell) REFERENCES NivellSuport (nivell)  
);
```

```
CREATE TABLE Tecnic (  
    mail VARCHAR2 (50 CHAR) CONSTRAINT NN_Tecnic_mail NOT NULL,  
    grup VARCHAR2 (20 CHAR) CONSTRAINT NN_Tecnic_grup NOT NULL,  
    CONSTRAINT PK_Tecnic PRIMARY KEY (mail),  
    CONSTRAINT FK_Tecnic_Usuari FOREIGN KEY (mail) REFERENCES Usuari (mail),  
    CONSTRAINT FK_Tecnic_Grup FOREIGN KEY (grup) REFERENCES GrupSuport (codi)  
);
```

```
CREATE TABLE Prioritat (
    prioritat INTEGER CONSTRAINT NN_Prioritat_prioritat NOT NULL,
    CONSTRAINT PK_Prioritat PRIMARY KEY (prioritat)
);
```

```
CREATE TABLE GrupServei (
    grup VARCHAR2 (20 CHAR) CONSTRAINT NN_GrupServei_grup NOT NULL,
    CONSTRAINT PK_GrupServei PRIMARY KEY (grup)
);
```

```
CREATE TABLE Servei (
    codi VARCHAR2 (20 CHAR) CONSTRAINT NN_Servei_codi NOT NULL,
    descripcio VARCHAR2 (50 CHAR),
    grup VARCHAR2 (20 CHAR) CONSTRAINT NN_Servei_grup NOT NULL,
    temps_mig INTEGER CONSTRAINT NN_Servei_temps NOT NULL,
    prioritat INTEGER CONSTRAINT NN_Servei_prioritat NOT NULL,
    CONSTRAINT PK_Servei PRIMARY KEY (codi),
    CONSTRAINT FK_Servei_Grup FOREIGN KEY (grup) REFERENCES GrupServei (grup),
    CONSTRAINT FK_Servei_Prioritat FOREIGN KEY (prioritat) REFERENCES Prioritat (prioritat)
);
```

```
CREATE TABLE OfertaGrupServei (
    grup VARCHAR2 (20 CHAR) CONSTRAINT NN_OfertaGrupServei_grup NOT NULL,
    servei VARCHAR2 (20 CHAR) CONSTRAINT NN_OfertaGrupServei_servei NOT NULL,
    CONSTRAINT PK_OfertaGrupServei PRIMARY KEY (grup, servei),
    CONSTRAINT FK_OfertaGrupServei_Grup FOREIGN KEY (grup) REFERENCES GrupSuport (codi),
    CONSTRAINT FK_OfertaGrupServei_Servei FOREIGN KEY (servei) REFERENCES Servei (codi)
);
```

```
CREATE TABLE TipusTicket (
    tipus VARCHAR2 (20 CHAR) CONSTRAINT NN_TipusTicket_tipus NOT NULL,
    CONSTRAINT PK_TipusTicket PRIMARY KEY (tipus)
);
```

```
CREATE TABLE Canal (
    canal VARCHAR2 (20 CHAR) CONSTRAINT NN_Canal_canal NOT NULL,
    CONSTRAINT PK_Canal PRIMARY KEY (canal)
);
```

```
CREATE TABLE CreacioTicket (
    metode VARCHAR2 (20 CHAR) CONSTRAINT NN_CreacioTicket_metode NOT NULL,
    CONSTRAINT PK_CreacioTicket PRIMARY KEY (metode)
);
```

```
CREATE TABLE Ticket (
    numero VARCHAR2 (20 CHAR) CONSTRAINT NN_Ticket_numero NOT NULL,
    data_hora_creacio TIMESTAMP CONSTRAINT NN_Ticket_data_hora_creacio NOT NULL,
    servei VARCHAR2 (20 CHAR) CONSTRAINT NN_Ticket_servei NOT NULL,
    prioritat INTEGER CONSTRAINT NN_Ticket_prioritat NOT NULL,
    tipus VARCHAR2 (20 CHAR) CONSTRAINT NN_Ticket_tipus NOT NULL,
    canal VARCHAR2 (20 CHAR) CONSTRAINT NN_Ticket_canal NOT NULL,
    creacio VARCHAR2 (20 CHAR) CONSTRAINT NN_Ticket_creacio NOT NULL,
    usuari VARCHAR2 (50 CHAR) CONSTRAINT NN_Ticket_usuari NOT NULL,
    CONSTRAINT PK_Ticket PRIMARY KEY (numero),
    CONSTRAINT FK_Ticket_Servei FOREIGN KEY (servei) REFERENCES Servei (codi),
    CONSTRAINT FK_Ticket_Prioritat FOREIGN KEY (prioritat) REFERENCES Prioritat (prioritat),
    CONSTRAINT FK_Ticket_TipusTicket FOREIGN KEY (tipus) REFERENCES TipusTicket (tipus),
    CONSTRAINT FK_Ticket_Canal FOREIGN KEY (canal) REFERENCES Canal (canal),
    CONSTRAINT FK_Ticket_CreacioTicket FOREIGN KEY (creacio) REFERENCES CreacioTicket (metode),
    CONSTRAINT FK_Ticket_Usuari FOREIGN KEY (usuari) REFERENCES Usuari (mail)
);
```

```

CREATE TABLE AssigTicketGrup (
    ticket VARCHAR2 (20 CHAR) CONSTRAINT NN_AssigTicketGrup_ticket NOT NULL,
    grup VARCHAR2 (20 CHAR) CONSTRAINT NN_AssigTicketGrup_grup NOT NULL,
    data_hora TIMESTAMP CONSTRAINT NN_AssigTicketGrup_data NOT NULL,
    CONSTRAINT PK_AssigTicketGrup PRIMARY KEY (ticket, grup),
    CONSTRAINT FK_AssigTicketGrup_Ticket FOREIGN KEY (ticket) REFERENCES Ticket (numero),
    CONSTRAINT FK_AssigTicketGrup_GrupSuport FOREIGN KEY (grup) REFERENCES GrupSuport (codi)
    /*RT1 ha de controlar-se al procediment d'alta d'una nova assignació*/
    /*RT5 ha de controlar-se al procediment d'alta d'una nova assignació*/
);

```

```

CREATE TABLE EstatTicket (
    estat VARCHAR2 (20 CHAR) CONSTRAINT NN_EstatTicket_estat NOT NULL,
    CONSTRAINT PK_EstatTicket PRIMARY KEY (estat)
);

```

```

CREATE TABLE CanviEstat (
    ticket VARCHAR2 (20 CHAR) CONSTRAINT NN_CanviEstat_ticket NOT NULL,
    data_hora TIMESTAMP CONSTRAINT NN_CanviEstat_data NOT NULL,
    tecnic VARCHAR2 (50 CHAR) CONSTRAINT NN_CanviEstat_tecnic NOT NULL,
    estat VARCHAR2 (20 CHAR) CONSTRAINT NN_CanviEstat_estat NOT NULL,
    CONSTRAINT PK_CanviEstat PRIMARY KEY (ticket, data_hora),
    CONSTRAINT FK_CanviEstat_ticket FOREIGN KEY (ticket) REFERENCES Ticket (numero),
    CONSTRAINT FK_CanviEstat_tecnic FOREIGN KEY (tecnic) REFERENCES Tecnic (mail),
    CONSTRAINT FK_CanviEstat_estat FOREIGN KEY (estat) REFERENCES EstatTicket (estat)
    /*RT2 ha de controlar-se al procediment d'alta d'un nou canvi d'estat*/
);

```

```

CREATE TABLE EstatSms (
    estat VARCHAR2 (20 CHAR) CONSTRAINT NN_EstatSms_estat NOT NULL,
    CONSTRAINT PK_EstatSms PRIMARY KEY (estat)
);

```

```

CREATE TABLE Missatge (
    codi VARCHAR2 (20 CHAR) CONSTRAINT NN_Missatge_codi NOT NULL,
    data_hora TIMESTAMP CONSTRAINT NN_Missatge_data NOT NULL,
    emissor VARCHAR2 (50 CHAR) CONSTRAINT NN_Missatge_emissor NOT NULL,
    estat VARCHAR2 (20 CHAR) CONSTRAINT NN_Missatge_estat NOT NULL,
    ticket VARCHAR2 (20 CHAR) CONSTRAINT NN_Missatge_ticket NOT NULL,
    CONSTRAINT PK_Missatge PRIMARY KEY (codi),
    CONSTRAINT FK_Missatge_estat FOREIGN KEY (estat) REFERENCES EstatSms (estat),
    CONSTRAINT FK_Missatge_ticket FOREIGN KEY (ticket) REFERENCES Ticket (numero)
    /*RT3 ha de controlar-se al procediment d'alta d'un nou missatge*/
);

```

```

CREATE TABLE Log (
    codi NUMBER CONSTRAINT NN_Log_codi NOT NULL,
    data_hora TIMESTAMP CONSTRAINT NN_Log_data NOT NULL,
    procediment VARCHAR2 (50 CHAR) CONSTRAINT NN_Log_procediment NOT NULL,
    param_in va_params CONSTRAINT NN_Log_param_in NOT NULL,
    RSP VARCHAR2 (200 CHAR) CONSTRAINT NN_Log_RSP NOT NULL,
    CONSTRAINT PK_Log PRIMARY KEY (codi)
);

```

11.3 Procediments d'alta, baixa i modificació de la Base de dades operativa

```
--PAIS
CREATE OR REPLACE PROCEDURE pais_alta (nom_pais IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO Pais
        VALUES (nom_pais);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. El pais ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'pais_alta', va_params(nom_pais), RSP);
    END;

END pais_alta;

/

CREATE OR REPLACE PROCEDURE pais_baixa (nom_pais IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM Pais
        WHERE nom = nom_pais;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Hi ha alguna seu ubicada en aquest pais';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'pais_baixa', va_params(nom_pais), RSP);
    END;

END pais_baixa;

--SEU
CREATE OR REPLACE PROCEDURE seu_alta (codi_seu IN VARCHAR2, direccio_seu IN VARCHAR2, tipologia_seu IN VARCHAR2, pais_seu IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO Seu
        VALUES (codi_seu, direccio_seu, tipologia_seu, pais_seu);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. La seu ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. A excepcio de la tipologia, tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. El pais no existeix';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'seu_alta', va_params(codi_seu, direccio_seu, tipologia_seu, pais_seu), RSP);
    END;

END seu_alta;

/

CREATE OR REPLACE PROCEDURE seu_baixa (codi_seu IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM Seu
        WHERE codi = codi_seu;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Existeixen usuaris inscrits en aquesta seu';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'seu_baixa', va_params(codi_seu), RSP);
    END;

END seu_baixa;
```



```

CREATE OR REPLACE PROCEDURE seu_modif_tipologia (codi_seu IN VARCHAR2, tipologia_seu IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificacio de la taula*/
    BEGIN
        --Update
        UPDATE Seu
        SET tipologia = tipologia_seu
        WHERE codi = codi_seu;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            RSP := 'ERROR: Error Inesperat';
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'seu_modif_tipologia', va_params(codi_seu, tipologia_seu), RSP);
    END;

END seu_modif_tipologia;

--USUARI
CREATE OR REPLACE PROCEDURE usuari_alta (mail_usuari IN VARCHAR2, seu_usuari IN VARCHAR2, RSP OUT VARCHAR2) AS
usuari_inactiu INTEGER;
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --verem si l'usuari ja existeix però este donat de baixa
        SELECT COUNT(*) INTO usuari_inactiu
        FROM Usuari
        WHERE mail = mail_usuari AND data_hora_baixa IS NOT NULL;

        --si es així, el tornem a donar d'alta
        IF usuari_inactiu > 0 THEN
            UPDATE Usuari
            SET data_hora_reg = CURRENT_TIMESTAMP, data_hora_baixa = NULL
            WHERE mail = mail_usuari;
        ELSE
            --si no, intentem inserir-lo a la base de dades
            INSERT INTO Usuari
            VALUES(mail_usuari, CURRENT_TIMESTAMP, NULL, seu_usuari);
        END IF;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. Aques usuari ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. A excepcio de la data de baixa, tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. La seu no existeix';
            ELSEIF SQLCODE = -02290 THEN RSP := 'ERROR: RT4. El valor de la data_baixa ha de ser nul o superior a la data_alta';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END IF;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'usuari_alta', va_params(mail_usuari, seu_usuari), RSP);
    END;

END usuari_alta;

/

CREATE OR REPLACE PROCEDURE usuari_baixa (mail_usuari IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        UPDATE Usuari
        SET data_hora_baixa = CURRENT_TIMESTAMP
        WHERE mail = mail_usuari;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            RSP := 'ERROR: Error Inesperat: ';
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'usuari_baixa', va_params(mail_usuari), RSP);
    END;

END usuari_baixa;

```

```

CREATE OR REPLACE PROCEDURE usuari_modif_seu (mail_usuari IN VARCHAR2, seu_usuari IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Usuari
        SET seu = seu_usuari
        WHERE mail = mail_usuari;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. La seu no pot ser nula';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. La seu no existeix';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'usuari_modif_seu', va_params(mail_usuari, seu_usuari), RSP);
    END;

END usuari_modif_seu;

/

--INTERACCIO
CREATE OR REPLACE PROCEDURE interaccio_alta (usuari_interaccio IN VARCHAR2, accio IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO Interaccio
        VALUES (usuari_interaccio, CURRENT_TIMESTAMP, accio);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquesta combinació (usuari - instant) ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'interaccio_alta', va_params(usuari_interaccio, accio), RSP);
    END;

END interaccio_alta;

```

```

--NIVELLSUPORT
CREATE OR REPLACE PROCEDURE nivellsuport_alta (nivell_nivellsuport IN VARCHAR2, valor_nivellsuport IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO Nivellsuport
        VALUES (nivell_nivellsuport, valor_nivellsuport);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. El nivell ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'nivellsuport_alta', va_params(nivell_nivellsuport), RSP);
    END;

END nivellsuport_alta;

/

CREATE OR REPLACE PROCEDURE nivellsuport_baixa (nivell_nivellsuport IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminació de la fila*/
    BEGIN
        --Delete
        DELETE FROM Nivellsuport
        WHERE nivell = nivell_nivellsuport;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02293 THEN RSP := 'ERROR: Violació clau forana. Hi ha alguna grup de suport al que correspon aquest nivell';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'nivellsuport_baixa', va_params(nivell_nivellsuport), RSP);
    END;

END nivellsuport_baixa;

```

```

--GRUPSUPORT
CREATE OR REPLACE PROCEDURE grupsuport_alta (codi_grupsuport IN VARCHAR2, nom_grupsuport IN VARCHAR2, nivell_grupsuport IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO GrupSuport
        VALUES(codi_grupsuport, nom_grupsuport, nivell_grupsuport);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. Aques grup ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. El nivell no existeix';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'grupsuport_alta',va_params(codi_grupsuport, nom_grupsuport, nivell_grupsuport),RSP);
    END;

END grupsuport_alta;

/

CREATE OR REPLACE PROCEDURE grupsuport_baixa (codi_grupsuport IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM GrupSuport
        WHERE codi = codi_grupsuport;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Aquest grup esta relacionat amb algun ticket o servei';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'grupsuport_baixa',va_params(codi_grupsuport),RSP);
    END;

END grupsuport_baixa;

/

CREATE OR REPLACE PROCEDURE grupsuport_modif_nom (codi_grupsuport IN VARCHAR2, nom_grupsuport IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificacio de la taula*/
    BEGIN
        --Update
        UPDATE GrupSuport
        SET nom = nom_grupsuport
        WHERE codi = codi_grupsuport;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El nom no pot ser nul';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'grupsuport_modif_nom',va_params(codi_grupsuport, nom_grupsuport),RSP);
    END;

END grupsuport_modif_nom;

/

--TECNIC
CREATE OR REPLACE PROCEDURE tecnic_alta (mail_tecnic IN VARCHAR2, grup_tecnic IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO Tecnic
        VALUES(mail_tecnic, grup_tecnic);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. Aques tecnic ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. El grup no existeix o el tecnic no es cap usuari';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'tecnic_alta',va_params(mail_tecnic, grup_tecnic),RSP);
    END;

END tecnic_alta;

```

```

CREATE OR REPLACE PROCEDURE tecnic_baixa (mail_tecnic IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminació de la taula*/
    BEGIN
        --Delete
        DELETE FROM Tecnic
        WHERE mail = mail_tecnic;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Aquest tecnic esta relacionat amb algun canvi d'estat d'un ticket';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
    END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP,'tecnic_baixa',va_params(mail_tecnic),RSP);
    END;

END tecnic_baixa;

/

CREATE OR REPLACE PROCEDURE tecnic_modif_grup (mail_tecnic IN VARCHAR2, grup_tecnic IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Tecnic
        SET grup = grup_tecnic
        WHERE mail = mail_tecnic;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El grup no pot ser nul';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. El grup no existeix';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
    END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP,'tecnic_modif_grup',va_params(mail_tecnic, grup_tecnic),RSP);
    END;

END tecnic_modif_grup;

/

--PRIORITAT
CREATE OR REPLACE PROCEDURE prioritat_alta (nivell_prioritat IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO Prioritat
        VALUES(nivell_prioritat);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primaria. La prioritat ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
    END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP,'prioritat_alta',va_params(nivell_prioritat),RSP);
    END;

END prioritat_alta;

/

CREATE OR REPLACE PROCEDURE prioritat_baixa (nivell_prioritat IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminació de la taula*/
    BEGIN
        --Delete
        DELETE FROM Prioritat
        WHERE prioritat = nivell_prioritat;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Hi ha algun ticket o servei amb aquesta prioritat';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
    END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP,'prioritat_baixa',va_params(nivell_prioritat),RSP);
    END;

END prioritat_baixa;

```

```

--GRUPSERVEI
CREATE OR REPLACE PROCEDURE grupservei_alta (grup_grupservei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO GrupServei
        VALUES (grup_grupservei);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. El grup ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'grupservei_alta', va_params (grup_grupservei), RSP);
    END;

END grupservei_alta;

/

CREATE OR REPLACE PROCEDURE grupservei_baixa (grup_grupservei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminació de la taula*/
    BEGIN
        --Delete
        DELETE FROM GrupServei
        WHERE grup = grup_grupservei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Hi ha alguna grup de suport que ofereix aquest grup de serveis';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'grupservei_baixa', va_params (grup_grupservei), RSP);
    END;

END grupservei_baixa;

/

--SERVEI
CREATE OR REPLACE PROCEDURE servei_alta (codi_servei IN VARCHAR2, descripcio_servei IN VARCHAR2, grup_servei IN VARCHAR2, temps_mig_servei IN INTEGER, prioritat_servei IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO Servei
        VALUES (codi_servei, descripcio_servei, grup_servei, temps_mig_servei, prioritat_servei);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquest servei ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. A excepció de la descripció, tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. El grup del servei o la prioritat no existeixen';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_alta', va_params (codi_servei, descripcio_servei, grup_servei, temps_mig_servei, prioritat_servei), RSP);
    END;

END servei_alta;

/

CREATE OR REPLACE PROCEDURE servei_baixa (codi_servei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminació de la taula*/
    BEGIN
        --Delete
        DELETE FROM Servei
        WHERE codi = codi_servei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Aquest servei està relacionat amb algun ticket';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la caïda al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_baixa', va_params (codi_servei), RSP);
    END;

END servei_baixa;

```

```

CREATE OR REPLACE PROCEDURE servei_modif_descripcio (codi_servei IN VARCHAR2, descripcio_servei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Servei
        SET descripcio = descripcio_servei
        WHERE codi = codi_servei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            RSP := 'ERROR: Error inesperat';
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_modif_descripcio', va_params(codi_servei, descripcio_servei), RSP);
    END;

    END servei_modif_descripcio;

/

CREATE OR REPLACE PROCEDURE servei_modif_grup (codi_servei IN VARCHAR2, grup_servei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Servei
        SET grup = grup_servei
        WHERE codi = codi_servei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El grup no pot ser nul';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. El grup no existeix';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_modif_grup', va_params(codi_servei, grup_servei), RSP);
    END;

    END servei_modif_grup;

/

CREATE OR REPLACE PROCEDURE servei_modif_temps (codi_servei IN VARCHAR2, temps_mig_servei IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Servei
        SET temps_mig = temps_mig_servei
        WHERE codi = codi_servei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El temps mig no pot ser nul';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_modif_temps', va_params(codi_servei, temps_mig_servei), RSP);
    END;

    END servei_modif_temps;

/

CREATE OR REPLACE PROCEDURE servei_modif_prioritat (codi_servei IN VARCHAR2, prioritat_servei IN INTEGER, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Servei
        SET prioritat = prioritat_servei
        WHERE codi = codi_servei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. LA prioritat no pot ser nula';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. LA prioritat no existeix';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'servei_modif_prioritat', va_params(codi_servei, prioritat_servei), RSP);
    END;

    END servei_modif_prioritat;

```

```

--OFERTAGRUPSERVEI
CREATE OR REPLACE PROCEDURE ofertagrupservei_alta (grup_ofertagrupservei IN VARCHAR2, servei_ofertagrupservei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO OfertaGrupServei
        VALUES(grup_ofertagrupservei, servei_ofertagrupservei);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. El conjunt (grup - servei) ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ofertagrupservei_alta', va_params(grup_ofertagrupservei, servei_ofertagrupservei), RSP);
    END;

END ofertagrupservei_alta;

/

CREATE OR REPLACE PROCEDURE ofertagrupservei_baixa (grup_ofertagrupservei IN VARCHAR2, servei_ofertagrupservei IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM OfertaGrupServei
        WHERE grup = grup_ofertagrupservei AND servei = servei_ofertagrupservei;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. El servei o el grup no existeixen';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ofertagrupservei_baixa', va_params(grup_ofertagrupservei, servei_ofertagrupservei), RSP);
    END;

END ofertagrupservei_baixa;

--TIPUSTICKET
CREATE OR REPLACE PROCEDURE tipusticket_alta (tipus_tipusticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO TipusTicket
        VALUES(tipus_tipusticket);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. El tipus ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'tipusticket_alta', va_params(tipus_tipusticket), RSP);
    END;

END tipusticket_alta;

/

CREATE OR REPLACE PROCEDURE tipusticket_baixa (tipus_tipusticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM TipusTicket
        WHERE tipus = tipus_tipusticket;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest tipus';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log
        VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'tipusticket_baixa', va_params(tipus_tipusticket), RSP);
    END;

END tipusticket_baixa;

```

```

--CANAL
CREATE OR REPLACE PROCEDURE canal_alta (tipus_canal IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO Canal
        VALUES (tipus_canal);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. El canal ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'canal_alta', va_params (tipus_canal), RSP);
    END;

END canal_alta;

/

CREATE OR REPLACE PROCEDURE canal_baixa (tipus_canal IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM Canal
        WHERE canal = tipus_canal;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest canal';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'canal_baixa', va_params (tipus_canal), RSP);
    END;

END canal_baixa;

--CREACIOTICKET
CREATE OR REPLACE PROCEDURE creacioticket_alta (metode_creacioticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Insercio a la taula*/
    BEGIN
        --Insert
        INSERT INTO CreacioTicket
        VALUES (metode_creacioticket);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violacio de clau primaria. El metode de creacio ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'creacioticket_alta', va_params (metode_creacioticket), RSP);
    END;

END creacioticket_alta;

/

CREATE OR REPLACE PROCEDURE creacioticket_baixa (metode_creacioticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Eliminacio de la taula*/
    BEGIN
        --Delete
        DELETE FROM CreacioTicket
        WHERE metode = metode_creacioticket;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest metode de creacio';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'creacioticket_baixa', va_params (metode_creacioticket), RSP);
    END;

END creacioticket_baixa;

```



```
--TICKET
CREATE OR REPLACE PROCEDURE ticket_alta (numero_ticket IN VARCHAR2, servei_ticket IN VARCHAR2, prioritat_ticket IN VARCHAR2, tipus_ticket IN VARCHAR2, canal_ticket IN VARCHAR2, creacio_ticket IN VARCHAR2, usuari_ticket IN VARCHAR2, RSP OUT VARCHAR2) AS
seu_dw VARCHAR2 (50 CHAR);
pais_dw VARCHAR2 (50 CHAR);
llista_canals_dw va_params;

BEGIN

/*Bloc 1: Inserció a la taula*/
BEGIN

--Insert
INSERT INTO Ticket
VALUES(numero_ticket, CURRENT_TIMESTAMP, servei_ticket, prioritat_ticket, tipus_ticket, canal_ticket, creacio_ticket, usuari_ticket);

RSP := 'OK';

/*Bloc 2: Crida als procediments del data warehouse*/
--tasques comues per a varies funcions del DW
SELECT seu_codi INTO seu_dw
FROM Seu INNER JOIN Usuari ON Usuari.seu = Seu.codi
WHERE Usuari.mail = usuari_ticket;

/*DW_usuari_mes_incidencies_INC (agrega)*/
IF (tipus_ticket = 'Incidencia') THEN
    DW_usuari_mes_incidencies_INC (usuari_ticket);
END IF;

/*DW_top10_peticions_seu_INC (aguja)*/
IF (tipus_ticket = 'Peticio') THEN
    DW_top10_peticions_seu_INC (seu_dw, servei_ticket);
END IF;

/*DW_servei_peticions_INC (agrega)*/
IF (tipus_ticket = 'Peticio') THEN
    DW_servei_peticions_INC (servei_ticket);
END IF;

/*DW_incidencies_seu_any_INC (aguja)*/
IF (tipus_ticket = 'Incidencia') THEN
    DW_incidencies_seu_any_INC (seu_dw);
END IF;

/*DW_sua_po_respostes_INC*/
DW_sua_po_respostes_INC;

/*DW_tickets_per_canal_INC (país, canal, nomu_canal(va_params))*/
--seleccionem el país on esta ubicada la seu al que pertany l'usuari
SELECT Seu.pais INTO pais_dw
FROM Seu INNER JOIN Usuari ON Usuari.seu = Seu.codi
WHERE Usuari.mail = usuari_ticket;

--seleccionem un llistat dels canals existents al sistema
SELECT canal BULK COLLECT INTO llista_canals_dw
FROM Canal;

DW_tickets_per_canal_INC(pais_dw, canal_ticket, llista_canals_dw);

--Tractament Excepcions
EXCEPTION
    WHEN OTHERS THEN
        IF SQLCODE = -9001 THEN RSP := 'ERROR: Violació de clau primaria. Aquest ticket ja existeix';
        ELSEIF SQLCODE = -0140 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
        ELSEIF SQLCODE = -02351 THEN RSP := 'ERROR: Violació de clau forana. El servei, prioritat, tipus, canal, creacio o usuari no existeixen';
        ELSE RSP := 'ERROR: Error inesperat';
        END IF;
END IF;

END;

/*Bloc 3: Registre de la crida al log*/
BEGIN
INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ticket_alta', va_params(numero_ticket, CURRENT_TIMESTAMP, servei_ticket, prioritat_ticket, tipus_ticket, canal_ticket, creacio_ticket, usuari_ticket), RSP);
END;

END ticket_alta;
```

```
CREATE OR REPLACE PROCEDURE ticket_baixa (numero_ticket IN VARCHAR2, RSP OUT VARCHAR2) AS
usuari_dw VARCHAR2 (50 CHAR);
seu_dw VARCHAR2 (50 CHAR);
servei_dw VARCHAR2 (50 CHAR);
pais_dw VARCHAR2 (50 CHAR);
canal_dw VARCHAR2 (50 CHAR);
tipus_dw VARCHAR2 (50 CHAR);

BEGIN

/*Bloc 1: Eliminació de la taula*/
BEGIN

--tasques comues per a varies funcions del DW
SELECT servei, tipus, canal, usuari INTO servei_dw, tipus_dw, canal_dw, usuari_dw
FROM Ticket
WHERE numero = numero_ticket;

SELECT seu INTO seu_dw
FROM Usuari
WHERE mail = usuari_dw;

SELECT pais INTO pais_dw
FROM Seu
WHERE codi = seu_dw;

--Delete
DELETE FROM Ticket
WHERE numero = numero_ticket;

RSP := 'OK';

/*Bloc 2: Crida als procediments del data warehouse*/

/*DW_usuari_mes_incidencies_DEC (aguja)*/
IF (tipus_dw = 'Incidencia') THEN
    DW_usuari_mes_incidencies_DEC (usuari_dw);
END IF;

/*DW_top10_peticions_seu_DEC (aguja)*/
IF (tipus_dw = 'Peticio') THEN
    DW_top10_peticions_seu_DEC (seu_dw, servei_dw);
END IF;

/*DW_servei_peticions_DEC (aguja)*/
IF (tipus_dw = 'Peticio') THEN
    DW_servei_peticions_DEC (servei_dw);
END IF;

/*DW_incidencies_seu_any_DEC (aguja)*/
IF (tipus_dw = 'Incidencia') THEN
    DW_incidencies_seu_any_DEC (seu_dw);
END IF;

/*DW_tickets_bancats_DEC*/
DW_tickets_bancats_DEC;

/*DW_tickets_per_canal_DEC (país, canal)*/
DW_tickets_per_canal_DEC(pais_dw, canal_dw);

--Tractament Excepcions
EXCEPTION
    WHEN OTHERS THEN
        IF SQLCODE = -02352 THEN RSP := 'ERROR: Violació clau forana. Aquest ticket esta relacionat amb algun canvi d'estat o grup de suport';
        ELSE RSP := 'ERROR: Error inesperat';
        END IF;
END IF;

END;

/*Bloc 3: Registre de la crida al log*/
BEGIN
INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ticket_baixa', va_params(numero_ticket), RSP);
END;

END ticket_baixa;
```

```

CREATE OR REPLACE PROCEDURE ticket_modif_canal (numero_ticket IN VARCHAR2, canal_ticket IN VARCHAR2, RSP OUT VARCHAR2) AS
canal_anterior VARCHAR2 (20 CHAR);
pais_ticket VARCHAR2 (50 CHAR);
BEGIN

    /*Bloc 1: Modificacio de la taula*/
    BEGIN

        --despen el canal anterior per modificar despres les taules del magatzem
        SELECT canal INTO canal_anterior
        FROM ticket
        WHERE numero = numero_ticket;

        --Update
        UPDATE Ticket
        SET canal = canal_ticket
        WHERE numero = numero_ticket;

        RSP := 'OK';

        /*Bloc 2: Crida als procediments del data warehouse*/
        /*DW_tickets_per_canal_MOD*/

        SELECT Seu.pais INTO pais_ticket
        FROM Ticket INNER JOIN Usuari ON Ticket.usuari = Usuari.mail
             INNER JOIN Seu ON Seu.codi = Usuari.seu
        WHERE Ticket.numero = numero_ticket;

        DW_tickets_per_canal_MOD(pais_ticket, canal_anterior, canal_ticket);

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El canal no pot ser nul';
            ELSIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. El canal no existeix';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;

    END;

    /*Bloc 3: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ticket_modif_canal', va_params(numero_ticket, canal_ticket), RSP);
    END;

END ticket_modif_canal;

/

CREATE OR REPLACE PROCEDURE ticket_modif_creacio (numero_ticket IN VARCHAR2, creacio_ticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN

    /*Bloc 1: Modificacio de la taula*/
    BEGIN

        --Update
        UPDATE Ticket
        SET creacio = creacio_ticket
        WHERE numero = numero_ticket;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. El metode de creacio no pot ser nul';
            ELSIF SQLCODE = -02291 THEN RSP := 'ERROR: Violacio de clau forana. El metode de creacio no existeix';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;

    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'ticket_modif_creacio', va_params(numero_ticket, creacio_ticket), RSP);
    END;

END;

```

```

--ASSIGTTICKETGRUP
CREATE OR REPLACE PROCEDURE assignticketgrup_alta (ticket_assignticketgrup IN VARCHAR2, grup_assignticketgrup IN VARCHAR2, RSP OUT VARCHAR2) AS
    nivell_actual INTEGER;
    nivell_futur INTEGER;
    oferta_servei INTEGER;
    data_assignacio_L3 TIMESTAMP;
    servei_no_ofert EXCEPTION;
    nivell_decrement EXCEPTION;
    nivell_dw VARCHAR2 (20 CHAR);
    tipus_dw VARCHAR2 (20 CHAR);

BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Restriccions generals
        --nivell del grup que es vol assignar
        SELECT nivellsupport.valor INTO nivell_futur
        FROM grupsupport INNER JOIN nivellsupport ON nivellsupport.nivell = grupsupport.nivell
        WHERE grupsupport.codi = grup_assignticketgrup;

        --nivell del grup que es actualment assignat
        SELECT MAX(nivellsupport.valor) INTO nivell_actual --assegurem el nivell del grup assignat actualment al ticket
        FROM assignticketgrup INNER JOIN grupsupport ON assignticketgrup.grup = grupsupport.codi
        INNER JOIN nivellsupport ON grupsupport.nivell = nivellsupport.nivell
        WHERE assignticketgrup.ticket = ticket_assignticketgrup
        ORDER BY assignticketgrup.data_hora DESC;

        IF nivell_futur < nivell_actual THEN
            RAISE nivell_decrement;
        END IF;

        --servei que sol·licita el ticket
        SELECT COUNT (*) INTO oferta_servei
        FROM ticket
        WHERE numero = ticket_assignticketgrup AND servei IN (
            SELECT servei
            FROM ofertagrupservei
            WHERE grup = grup_assignticketgrup
        );

        IF oferta_servei < 1 THEN
            RAISE servei_no_ofert;
        END IF;

        --Insert
        INSERT INTO AssignTicketGrup
        VALUES(ticket_assignticketgrup, grup_assignticketgrup, CURRENT_TIMESTAMP);

        RSP := 'OK';

        /*Bloc 2: Crida als procediments del data warehouse*/

        /*DW_hores_tickets_L3_INC*/
        SELECT nivell INTO nivell_dw
        FROM grupsupport
        WHERE codi = grup_assignticketgrup;

        IF (nivell_dw = 'Problem Management') THEN
            --el grup que es de nivell 'Problem management'. Comptem les hores que el ticket ha passat amb el grup L3
            SELECT data_hora INTO data_assignacio_L3
            FROM (
                SELECT data_hora, rownum rnum
                FROM (
                    SELECT data_hora
                    FROM assignticketgrup
                    WHERE ticket = ticket_assignticketgrup
                    ORDER BY data_hora DESC
                )
            )
            WHERE rnum = 3;

            DW_hores_tickets_L3_INC(data_assignacio_L3);

            /*DW_incidencies_PM_INC*/
            SELECT tipus INTO tipus_dw
            FROM ticket
            WHERE numero = ticket_assignticketgrup;
            IF (tipus_dw = 'Incidencia') THEN
                --l'assignació es a un grup PM i el ticket es de tipus incidència
                DW_incidencies_PM_INC;
            END IF;
        END IF;

        --Tractament Excepcions
        EXCEPTION
        WHEN nivell_decrement THEN
            RSP := 'ERROR: Restricció textual. El nivell del grup que s'intenta assignar al ticket es inferior al nivell del grup que el ticket ja te assignat actualment';
        WHEN servei_no_ofert THEN
            RSP := 'ERROR: Restricció textual. El grup que s'intenta assignar al ticket no ofereix el servei que el ticket demana';
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquesta assignació ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSEIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. El grup o el ticket no existeixen';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END;

        /*Bloc 3: Registre de la crida al Log*/
        BEGIN
            INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'assignticketgrup_alta', va_params(ticket_assignticketgrup, grup_assignticketgrup), RSP);
        END;

    END assignticketgrup_alta;

```

```

CREATE OR REPLACE PROCEDURE assigticketgrup_baixa (ticket_assigticketgrup IN VARCHAR2, grup_assigticketgrup IN VARCHAR2, RSP OUT VARCHAR2) AS
nivell_dw VARCHAR2 (30 CHAR);
tipus_dw VARCHAR2 (20 CHAR);
BEGIN
    /*Bloc 1: Eliminació de la taula*/
    BEGIN
        --Delete
        DELETE FROM AssigTicketGrup
        WHERE ticket = ticket_assigticketgrup AND grup = grup_assigticketgrup;

        RSP := 'OK';

        /*Bloc 2: Crida als procediments del data warehouse*/
        /*DW_incidencies_PM_DEC*/
        SELECT nivell INTO nivell_dw
        FROM grupsupport
        WHERE codi = grup_assigticketgrup;

        SELECT tipus INTO tipus_dw
        FROM ticket
        WHERE numero = ticket_assigticketgrup;

        IF (nivell_dw = 'Problem Management' AND tipus_dw = 'Incidencia') THEN
            DW_incidencies_PM_DEC;
        END IF;

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            RSP := 'ERROR: Error Inesperat';
        END;

        /*Bloc 3: Registre de la crida al Log*/
        BEGIN
            INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'assigticketgrup_baixa', va_params(ticket_assigticketgrup, grup_assigticketgrup), RSP);
        END;
    END assigticketgrup_baixa;
/

--ESTATTICKET
CREATE OR REPLACE PROCEDURE estatsticket_alta (estat_estatticket IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Injecció a la taula*/
    BEGIN
        --Insert
        INSERT INTO EstatTicket
        VALUES(estat_estatticket);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -90001 THEN RSP := 'ERROR: Violació de clau primària. Aquest estat ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END;

        /*Bloc 2: Registre de la crida al Log*/
        BEGIN
            INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'estatsticket_alta', va_params(estat_estatticket), RSP);
        END;
    END estatsticket_alta;

    /*Bloc 1: Injecció a la taula*/
    BEGIN
        --Insert
        DELETE FROM EstatTicket
        WHERE estat = estat_estatticket;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Aquest estat està relacionat amb algun ticket';
            ELSE RSP := 'ERROR: Error Inesperat';
            END IF;
        END;

        /*Bloc 2: Registre de la crida al Log*/
        BEGIN
            INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'estatsticket_baixa', va_params(estat_estatticket), RSP);
        END;
    END estatsticket_baixa;

```

```

--CANVIESTAT
CREATE OR REPLACE PROCEDURE canviestat_alta (ticket_canviestat IN VARCHAR2, tecnic_canviestat IN VARCHAR2, estat_canviestat IN VARCHAR2, RSP OUT VARCHAR2) AS
grup_tecnic VARCHAR2 (20 CHAR);
grup_actual_ticket VARCHAR2 (20 CHAR);
tecnic_prohibit EXCEPTION;
tipus_dw VARCHAR2 (20 CHAR);
temps_mig_dw INTEGER;
nivell_grup_dw VARCHAR (20 CHAR);
data_assignacio_l3 TIMESTAMP;
data_creacio_dw TIMESTAMP;
usuari_dw VARCHAR2 (20 CHAR);
prioritat_dw INTEGER;
seu_dw VARCHAR2 (20 CHAR);
servei_dw VARCHAR2 (20 CHAR);
pais_dw VARCHAR2 (20 CHAR);
canal_dw VARCHAR (20 CHAR);

BEGIN
    /*Bloc 1: Connexió a la base de dades*/
    BEGIN
        --Restriccions generals
        --SEU AL QUE PERTANY EL TECNIC QUE VOL EFECTUAR EL CANVI D'ESTAT
        SELECT grup INTO grup_tecnic
        FROM tecnic
        WHERE mail = tecnic_canviestat;

        --SEU ASSIGNAT ACTUALMENT AL TICKET
        SELECT grup INTO grup_actual_ticket
        FROM (
            SELECT grup
            FROM assignatictgrup
            WHERE ticket = ticket_canviestat
            ORDER BY data_hora DESC
        )
        WHERE ROWNUM = 1;

        --el TECNIC QUE AUTENTICA EFECTUA EL CANVI D'ESTAT, NO ESTE ASSIGNAT AL TICKET ACTUALMENT
        IF grup_tecnic != grup_actual_ticket THEN
            RAISE tecnic_prohibit;
        END IF;

        --Insert
        INSERT INTO CanviEstat
        VALUES(ticket_canviestat, CURRENT_TIMESTAMP, tecnic_canviestat, estat_canviestat);

        RSP := 'OK';

        /*Bloc 2: Crida als procediments del data warehouse*/

        --PARÀMETRES COMUNS PER A VARIES FUNCIONS DEL DW
        --(tipus_dw, data_creacio, usuari, seu, servei, tipus, canal i país)
        SELECT prioritat, data_hora_creacio, usuari, servei, tipus, canal
        INTO prioritat_dw, data_creacio_dw, usuari_dw, servei_dw, tipus_dw, canal_dw
        FROM Ticket
        WHERE numero = ticket_canviestat;

        SELECT seu INTO seu_dw
        FROM Usuari
        WHERE mail = usuari_dw;

        SELECT país INTO país_dw
        FROM Seu
        WHERE codi = seu_dw;

        /*DW_temps_incidentis_prio_INC (prioritat, data_creacio)*/
        IF (tipus_dw = 'Incidencia' AND estat_canviestat = 'Resolt') THEN
            DW_temps_incidentis_prio_INC (prioritat_dw, data_creacio_dw);
        END IF;

        /*DW_peticions_sobre_temps_INC (temps_mig, data_creacio, país)*/
        IF (tipus_dw = 'Peticio' AND estat_canviestat = 'Resolt') THEN
            --temps_mig de assignat al ticket
            SELECT Servei.temps_mig INTO temps_mig_dw
            FROM Ticket INNER JOIN Servei ON Ticket.servei = Servei.codi
            WHERE Ticket.numero = ticket_canviestat;

            DW_peticions_sobre_temps_INC (temps_mig_dw, data_creacio_dw, país_dw);
        END IF;

        /*DW_usuari_mes_incidentis_DEC (usuari)*/
        IF (tipus_dw = 'Incidencia' AND estat_canviestat = 'Cancel·lat') THEN
            DW_usuari_mes_incidentis_DEC (usuari_dw);
        END IF;

        /*DW_top10_peticions_seu_DEC (seu, servei)*/
        IF (tipus_dw = 'Peticio' AND estat_canviestat = 'Cancel·lat') THEN
            DW_top10_peticions_seu_DEC (seu_dw, servei_dw);
        END IF;

        /*DW_tickets_resolta_mes_INC*/
        IF (estat_canviestat = 'Resolt') THEN
            DW_tickets_resolta_mes_INC;
        END IF;

        /*DW_servei_peticions_DEC (servei)*/
        IF (tipus_dw = 'Peticio' AND estat_canviestat = 'Cancel·lat') THEN
            DW_servei_peticions_DEC (servei_dw);
        END IF;

        /*DW_incidentis_seu_any_DEC (seu)*/
        IF (tipus_dw = 'Incidencia' AND estat_canviestat = 'Cancel·lat') THEN
            DW_incidentis_seu_any_DEC (seu_dw);
        END IF;

        /*DW_hores_tickets_l3_INC*/
        SELECT nivell INTO nivell_grup_dw
        FROM GrupReport
        WHERE codi = grup_actual_ticket;

        IF (estat_canviestat = 'Resolt' AND nivell_grup_dw = 'L3') THEN
            SELECT data_hora INTO data_assignacio_l3
            FROM (
                SELECT data_hora
                FROM assignatictgrup
                WHERE ticket = ticket_canviestat
                ORDER BY data_hora DESC
            )
            WHERE ROWNUM = 1;

            DW_hores_tickets_l3_INC(data_assignacio_l3);
        END IF;

        /*DW_tickets_tancats_INC*/
        IF (estat_canviestat = 'Resolt' OR estat_canviestat = 'Cancel·lat') THEN
            DW_tickets_tancats_INC (seu_dw);
        END IF;

        /*DW_tickets_per_canal_DEC (país, canal)*/
        IF (estat_canviestat = 'Cancel·lat') THEN
            DW_tickets_per_canal_DEC (país_dw, canal_dw);
        END IF;

        --Tractament d'errors
        EXCEPTION
        WHEN tecnic_prohibit THEN
            RSP := 'ERROR: Restricció tental. El tecnic no pertany al grup que actualment s'encarrega del ticket';

        WHEN OTHERS THEN
            IF SQLCODE = -9001 THEN RSP := 'ERROR: Violació de clau primària. Aquest canvi d'estat ja existeix';
            ELSEIF SQLCODE = -91400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;

        END;

        /*Bloc 3: Registre de la crida al Log*/
        BEGIN
            INSERT INTO Log VALUES(log_seq.nextval, CURRENT_TIMESTAMP, 'canviestat_alta', va_param(ticket_canviestat, tecnic_canviestat, estat_canviestat), RSP);
        END;

    END canviestat_alta;

```

```

--ESTATSMs
CREATE OR REPLACE PROCEDURE estatstms_alta (estat_estatstms IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        INSERT INTO Estatstms
        VALUES (estat_estatstms);

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquest estat ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'estatstms_alta', va_params (estat_estatstms), RSP);
    END;

END estatstms_alta;

/

--ESTATSMs baixa
CREATE OR REPLACE PROCEDURE estatstms_baixa (estat_estatstms IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Insert
        DELETE FROM Estatstms
        WHERE estat = estat_estatstms;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -02292 THEN RSP := 'ERROR: Violació clau forana. Aquest estat està relacionat amb algun missatge';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'estatstms_baixa', va_params (estat_estatstms), RSP);
    END;

END estatstms_baixa;

--MISSATGE
CREATE OR REPLACE PROCEDURE missatge_alta (codi_missatge IN VARCHAR2, emissor_missatge IN VARCHAR2, estat_missatge IN VARCHAR2, ticket_missatge IN VARCHAR2, RSP OUT VARCHAR2) AS
usuari_diferent EXCEPTION;
usuari_ticket VARCHAR2 (50 CHAR);
qtb_sms_usuari_dw INTEGER;

BEGIN
    /*Bloc 1: Inserció a la taula*/
    BEGIN
        --Restriccions SQL
        IF emissor_missatge != 'SYS' THEN
            --seleccionem l'usuari que ha generat el ticket associat al missatge
            SELECT usuari INTO usuari_ticket
            FROM ticket
            WHERE numero = ticket_missatge;

            IF usuari_ticket != emissor_missatge THEN
                RAISE usuari_diferent;
            END IF;
        END IF;

        --Insert
        INSERT INTO Missatge VALUES (codi_missatge, CURRENT_TIMESTAMP, emissor_missatge, estat_missatge, ticket_missatge);

        RSP := 'OK';

        /*Bloc 2: Crida als procediments del data warehouse*/
        /*DW_sms_no_respostes_DEC*/
        SELECT COUNT (*) INTO qtb_sms_usuari_dw
        FROM missatge
        WHERE ticket = ticket_missatge AND emissor != 'SYS';
        IF (emissor_missatge != 'SYS' AND qtb_sms_usuari_dw = 1) THEN
            --L'emissor es l'usuari i es el primer missatge que no ha estat enviat per sys. Decrementem el nombre de tickets no respostes
            DW_sms_no_respostes_DEC;
        END IF;

        --Tractament Excepcions
        EXCEPTION
        WHEN usuari_diferent THEN
            RSP := 'L'emissor del missatge ha de ser SYS o l'usuari al que correspon el ticket';

        WHEN OTHERS THEN
            IF SQLCODE = -00001 THEN RSP := 'ERROR: Violació de clau primària. Aquest missatge ja existeix';
            ELSEIF SQLCODE = -01400 THEN RSP := 'ERROR: Null. Tots els atributs han de tenir algun valor';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
        END IF;
    END;

    /*Bloc 3: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES (Log_seq.nextval, CURRENT_TIMESTAMP, 'missatge_alta', va_params (codi_missatge, emissor_missatge, estat_missatge, ticket_missatge), RSP);
    END;

END missatge_alta;

```

```

CREATE OR REPLACE PROCEDURE missatge_modif_estat (codi_missatge IN VARCHAR2, estat_missatge IN VARCHAR2, RSP OUT VARCHAR2) AS
BEGIN
    /*Bloc 1: Modificació de la taula*/
    BEGIN
        --Update
        UPDATE Missatge
        SET estat = estat_missatge
        WHERE codi = codi_missatge;

        RSP := 'OK';

        --Tractament Excepcions
        EXCEPTION
        WHEN OTHERS THEN
            IF SQLCODE = -01407 THEN RSP := 'ERROR: Null. L'estat no pot ser nul';
            ELSIF SQLCODE = -02291 THEN RSP := 'ERROR: Violació de clau forana. L'estat no existeix';
            ELSE RSP := 'ERROR: Error inesperat';
            END IF;
    END;

    /*Bloc 2: Registre de la crida al Log*/
    BEGIN
        INSERT INTO Log VALUES(Log_seq.nextval, CURRENT_TIMESTAMP, 'missatge_modif_estat', va_params(codi_missatge, estat_missatge), RSP);
    END;
END missatge_modif_estat;

```

11.4 Taules del Data Warehouse

```
CREATE TABLE DW_temps_incidencies_prio (  
    mes INTEGER NOT NULL,  
    ani INTEGER NOT NULL,  
    prioritat INTEGER NOT NULL,  
    qtt_incidencies INTEGER NOT NULL,  
    temps_total INTEGER NOT NULL,  
    CONSTRAINT PK_DW_temps_incidencies_prio PRIMARY KEY (mes, ani, prioritat)  
);
```

```
CREATE TABLE DW_peticions_sobre_temps (  
    mes INTEGER NOT NULL,  
    ani INTEGER NOT NULL,  
    pais VARCHAR2 (50 CHAR) NOT NULL,  
    qtt_peticions INTEGER NOT NULL,  
    qtt_peticions_sobre_temps INTEGER NOT NULL,  
    CONSTRAINT PK_DW_peticions_sobre_temps PRIMARY KEY (mes, ani, pais)  
);
```

```
CREATE TABLE DW_usuari_mes_incidencies (  
    usuari VARCHAR2 (50 CHAR) NOT NULL,  
    qtt_incidencies INTEGER NOT NULL,  
    CONSTRAINT PK_DW_usuari_mes_incidencies PRIMARY KEY (usuari)  
);
```

```
CREATE TABLE DW_top10_peticions_seu (  
    seu VARCHAR2 (20 CHAR) NOT NULL,  
    servei VARCHAR2 (20 CHAR) NOT NULL,  
    qtt_peticions INTEGER NOT NULL,  
    CONSTRAINT PK_DW_top10_peticions_seu PRIMARY KEY (seu, servei)  
);
```

```
CREATE TABLE DW_tickets_resolts_mes (  
    mes INTEGER NOT NULL,  
    ani INTEGER NOT NULL,  
    qtt_peticions INTEGER NOT NULL,  
    CONSTRAINT PK_DW_tickets_resolts_mes PRIMARY KEY (mes, ani)  
);
```

```
CREATE TABLE DW_servei_peticions (  
    servei VARCHAR2 (20 CHAR) NOT NULL,  
    qtt_peticions INTEGER NOT NULL,  
    CONSTRAINT PK_DW_servei_peticions PRIMARY KEY (servei)  
);
```

```
CREATE TABLE DW_incidencies_seu_any (  
    ani INTEGER NOT NULL,  
    seu VARCHAR2 (20 CHAR) NOT NULL,  
    qtt_incidencies INTEGER NOT NULL,  
    CONSTRAINT PK_DW_incidencies_seu_any PRIMARY KEY (ani, seu)  
);
```



```
CREATE TABLE DW_hores_tickets_L3 (  
    ani INTEGER NOT NULL,  
    qtt_hores INTEGER NOT NULL,  
    CONSTRAINT PK_DW_hores_tickets_L3 PRIMARY KEY (ani)  
);
```

```
CREATE TABLE DW_incidencies_PM (  
    ani INTEGER NOT NULL,  
    qtt_incidencies INTEGER NOT NULL,  
    CONSTRAINT PK_DW_incidencies_PM PRIMARY KEY (ani)  
);
```

```
CREATE TABLE DW_sms_no_respostos (  
    ani INTEGER NOT NULL,  
    qtt_tickets INTEGER NOT NULL,  
    qtt_tickets_no_respostos INTEGER NOT NULL,  
    CONSTRAINT PK_DW_sms_no_respostos PRIMARY KEY (ani)  
);
```

```
CREATE TABLE DW_tickets_tancats (  
    qtt_tancats INTEGER NOT NULL,  
    CONSTRAINT PK_DW_tickets_tancats PRIMARY KEY (qtt_tancats)  
);
```

```
CREATE TABLE DW_tickets_per_canal (  
    pais VARCHAR2 (50 CHAR) NOT NULL,  
    mes INTEGER NOT NULL,  
    ani INTEGER NOT NULL,  
    canal VARCHAR2 (20 CHAR) NOT NULL,  
    qtt_tickets_canal INTEGER NOT NULL,  
    qtt_tickets_total INTEGER NOT NULL,  
    CONSTRAINT PK_DW_tickets_per_canal PRIMARY KEY (pais, mes, ani, canal)  
);
```

11.5 Procediments d'actualització del Data Warehouse

```
CREATE OR REPLACE PROCEDURE DW_temps_incidencies_prio_INC (prioritat_in IN INTEGER, data_creacio_in IN TIMESTAMP) AS
temps_transcorregut INTEGER;

BEGIN
--calculem el temps transcorregut (en segons) des de la creació fins aquest moment
temps_transcorregut := ((CAST (CURRENT_TIMESTAMP AS DATE) - CAST (data_creacio_in AS DATE)) * 86400);

--UPDATE a DW_temps_incidencies_prio
UPDATE DW_temps_incidencies_prio
SET qtt_incidencies = qtt_incidencies+1, temps_total = temps_total + temps_transcorregut
WHERE ani = to_char(sysdate, 'YYYY') AND mes = to_char(sysdate, 'MM') AND prioritat = prioritat_in;

--INSERT INTO en cas que el registre no existís prèviament
IF sql%rowcount = 0 THEN
    INSERT INTO DW_temps_incidencies_prio
    VALUES (to_char(sysdate, 'MM'), to_char(sysdate, 'YYYY'), prioritat_in, 1, temps_transcorregut);
END IF;

END DW_temps_incidencies_prio_INC;

BEGIN
--calculem el temps transcorregut (en segons) des de la creació fins aquest moment
temps_transcorregut := ((CAST (CURRENT_TIMESTAMP AS DATE) - CAST (data_creacio_in AS DATE)) * 86400);

--UPDATE a DW_peticions_sobre_temps
UPDATE DW_peticions_sobre_temps
SET qtt_peticions = qtt_peticions+1
WHERE ani = to_char(sysdate, 'YYYY') AND mes = to_char(sysdate, 'MM') AND pais = pais_in;

--INSERT INTO en cas que el registre no existís prèviament
IF sql%rowcount = 0 THEN
    INSERT INTO DW_peticions_sobre_temps
    VALUES (to_char(sysdate, 'MM'), to_char(sysdate, 'YYYY'), pais_in, 1, 0);
END IF;

--en cas que el temps transcorregut sigui superior al temps mig de resolució, augmentem
IF temps_transcorregut > temps_mig_in THEN
    UPDATE DW_peticions_sobre_temps
    SET qtt_peticions_sobre_temps = qtt_peticions_sobre_temps+1
    WHERE ani = to_char(sysdate, 'YYYY') AND mes = to_char(sysdate, 'MM') AND pais = pais_in;
END IF;

END DW_peticions_sobre_temps_INC;

CREATE OR REPLACE PROCEDURE DW_usuari_mes_incidencies_INC (usuari_in IN VARCHAR2) AS
BEGIN
--UPDATE a DW_usuari_mes_incidencies
UPDATE DW_usuari_mes_incidencies
SET qtt_incidencies = qtt_incidencies+1
WHERE usuari = usuari_in;

--INSERT INTO en cas que el registre no existís prèviament
IF sql%rowcount = 0 THEN
    INSERT INTO DW_usuari_mes_incidencies
    VALUES (usuari_in, 1);
END IF;

END DW_usuari_mes_incidencies_INC;

CREATE OR REPLACE PROCEDURE DW_usuari_mes_incidencies_DEC (usuari_in IN VARCHAR2) AS
BEGIN
--UPDATE a DW_usuari_mes_incidencies
UPDATE DW_usuari_mes_incidencies
SET qtt_incidencies = qtt_incidencies-1
WHERE usuari = usuari_in;

END DW_usuari_mes_incidencies_DEC;
```

```

CREATE OR REPLACE PROCEDURE DW_top10_peticions_seu_INC (seu_in IN VARCHAR2, servei_in IN VARCHAR2) AS
BEGIN
    --UPDATE a DW_top10_peticions_seu
    UPDATE DW_top10_peticions_seu
    SET qtt_peticions = qtt_peticions+1
    WHERE seu = seu_in AND servei = servei_in;

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_top10_peticions_seu
        VALUES (seu_in, servei_in, 1);
    END IF;
END DW_top10_peticions_seu_INC;

```

```

CREATE OR REPLACE PROCEDURE DW_top10_peticions_seu_DEC (seu_in IN VARCHAR2, servei_in IN VARCHAR2) AS
BEGIN
    --UPDATE a DW_top10_peticions_seu
    UPDATE DW_top10_peticions_seu
    SET qtt_peticions = qtt_peticions-1
    WHERE seu = seu_in AND servei = servei_in;
END DW_top10_peticions_seu_DEC;

```

```

CREATE OR REPLACE PROCEDURE DW_tickets_resolts_mes_INC AS
BEGIN
    --UPDATE a DW_usuari_mes_incidencies
    UPDATE DW_tickets_resolts_mes
    SET qtt_peticions = qtt_peticions+1
    WHERE ani = to_char(sysdate, 'YYYY') AND mes = to_char(sysdate, 'MM');

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_tickets_resolts_mes
        VALUES (to_char(sysdate, 'MM'), to_char(sysdate, 'YYYY'), 1);
    END IF;
END DW_tickets_resolts_mes_INC;

```

```

CREATE OR REPLACE PROCEDURE DW_servei_peticions_INC (servei_in IN VARCHAR2)AS
BEGIN
    --UPDATE a DW_usuari_mes_incidencies
    UPDATE DW_servei_peticions
    SET qtt_peticions = qtt_peticions+1
    WHERE servei = servei_in;

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_servei_peticions
        VALUES (servei_in, 1);
    END IF;
END DW_servei_peticions_INC;

```

```

CREATE OR REPLACE PROCEDURE DW_servei_peticions_DEC (servei_in IN VARCHAR2)AS
BEGIN
    --UPDATE a DW_usuari_mes_incidencies
    UPDATE DW_servei_peticions
    SET qtt_peticions = qtt_peticions-1
    WHERE servei = servei_in;
END DW_servei_peticions_DEC;

```

```

CREATE OR REPLACE PROCEDURE DW_incidencies_seu_any_INC (seu_in IN VARCHAR2)AS
BEGIN

    --UPDATE a DW_usuari_mes_incidencies
    UPDATE DW_incidencies_seu_any
    SET qtt_incidencies = qtt_incidencies+1
    WHERE ani = to_char(sysdate, 'YYYY') AND seu = seu_in;

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_incidencies_seu_any
        VALUES (to_char(sysdate, 'YYYY'), seu_in, 1);
    END IF;

END DW_incidencies_seu_any_INC;

```

```

CREATE OR REPLACE PROCEDURE DW_incidencies_seu_any_DEC (seu_in IN VARCHAR2)AS
BEGIN

    --UPDATE a DW_usuari_mes_incidencies
    UPDATE DW_incidencies_seu_any
    SET qtt_incidencies = qtt_incidencies-1
    WHERE ani = to_char(sysdate, 'YYYY') AND seu = seu_in;

END DW_incidencies_seu_any_DEC;

```

```

CREATE OR REPLACE PROCEDURE DW_hores_tickets_L3_INC (data_assig_L3_in IN TIMESTAMP) AS
hores_invertides INTEGER;
BEGIN

    --calculen el temps transcorregut (en hores) des de l'assignacio al grup L3 fins ara
    hores_invertides := ((CAST (CURRENT_TIMESTAMP AS DATE) - CAST (data_assig_L3_in AS DATE)) * 3600);

    --UPDATE a DW_hores_tickets_L3
    UPDATE DW_hores_tickets_L3
    SET qtt_hores = qtt_hores + hores_invertides;
    WHERE ani = to_char(sysdate, 'YYYY');

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_hores_tickets_L3
        VALUES (to_char(sysdate, 'YYYY'), hores_invertides);
    END IF;

END DW_hores_tickets_L3_INC;

```

```

CREATE OR REPLACE PROCEDURE DW_incidencies_PM_INC AS
BEGIN

    --UPDATE a DW_incidencies_PM
    UPDATE DW_incidencies_PM
    SET qtt_incidencies = qtt_incidencies+1;
    WHERE ani = to_char(sysdate, 'YYYY');

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_incidencies_PM
        VALUES (to_char(sysdate, 'YYYY'), 1)
    END IF;

END DW_incidencies_PM_INC;

```

```
CREATE OR REPLACE PROCEDURE DW_incidencias_PM_INC AS
BEGIN
```

```

    --UPDATE a DW_incidencias_PM
    UPDATE DW_incidencias_PM
    SET qtt_incidencias = qtt_incidencias+1;
    WHERE ani = to_char(sysdate, 'YYYY');

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_incidencias_PM
        VALUES (to_char(sysdate, 'YYYY'), 1)
    END IF;

```

```
END DW_incidencias_PM_INC;
```

```
CREATE OR REPLACE PROCEDURE DW_sms_no_respostos_INC AS
BEGIN
```

```

    --UPDATE a DW_sms_no_respostos
    UPDATE DW_sms_no_respostos
    SET qtt_tickets = qtt_tickets+1, qtt_tickets_no_respostos = qtt_tickets_no_respostos+1
    WHERE ani = to_char(sysdate, 'YYYY');

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_sms_no_respostos
        VALUES (to_char(sysdate, 'YYYY'), 1, 1);
    END IF;

```

```
END DW_sms_no_respostos_INC;
```

```
CREATE OR REPLACE PROCEDURE DW_sms_no_respostos_DEC AS
BEGIN
```

```

    --UPDATE a DW_sms_no_respostos
    UPDATE DW_sms_no_respostos
    SET qtt_tickets_no_respostos = qtt_incidencias_no_respostos-1
    WHERE ani = to_char(sysdate, 'YYYY');

```

```
END DW_sms_no_respostos_DEC;
```

```
CREATE OR REPLACE PROCEDURE DW_tickets_tancats_INC AS
BEGIN
```

```

    --UPDATE a DW_tickets_tancats
    UPDATE DW_tickets_tancats
    SET qtt_tickets = qtt_tickets+1;

    --INSERT INTO en cas que el registre no existis previamnet
    IF sql%rowcount = 0 THEN
        INSERT INTO DW_tickets_tancats
        VALUES (1);
    END IF;

```

```
END DW_tickets_tancats_INC;
```

```

CREATE OR REPLACE PROCEDURE DW_tickets_tancats_DEC AS
BEGIN
    --UPDATE a DW_tickets_tancats
    UPDATE DW_tickets_tancats
    SET qtt_tickets = qtt_tickets-1;
END DW_tickets_tancats_DEC;

CREATE OR REPLACE PROCEDURE DW_tickets_per_canal_INC (pais_in IN VARCHAR2, canal_in IN VARCHAR2, noms_canals_in IN va_params)AS
existeix INTEGER;
BEGIN
    --comproven si existeixen les entrades per aquell any/mes/pais
    SELECT COUNT(*) INTO existeix
    FROM DW_tickets_per_canal
    WHERE ani = to_char(sysdate, 'YYYY') AND mes = to_char(sysdate, 'MM') AND pais = pais_in;

    IF existeix > 0 THEN
        --les entrades existeixen
        --+1 a qtt_tickets_total per aquell pais/any/mes
        UPDATE DW_tickets_per_canal
        SET qtt_tickets_total = qtt_tickets_total+1
        WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY');
    ELSE
        --no existeix l'entrada
        FOR n IN 1..noms_canals_in.count
        --creem tantes entrades com canals hi hagi al sistema, amb qtt_tickets_total = 1
        LOOP
            INSERT INTO DW_tickets_per_canal
            VALUES (pais_in, to_char(sysdate, 'MM'), to_char(sysdate, 'YYYY'), noms_canals_in(n), 0, 1);
        END LOOP;
    END IF;

    --finalment incrementem en 1 el canal pel que s'ha creat el ticket
    UPDATE DW_tickets_per_canal
    SET qtt_tickets_canal = qtt_tickets_canal+1
    WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY') AND canal = canal_in;
END DW_tickets_per_canal_INC;

CREATE OR REPLACE PROCEDURE DW_tickets_per_canal_DEC (pais_in IN VARCHAR2, canal_in IN VARCHAR2)AS
BEGIN
    -- -1 a qtt_tickets_total per aquell pais/any/mes
    UPDATE DW_tickets_per_canal
    SET qtt_tickets_total = qtt_tickets_total-1
    WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY');

    -- -1 pel canal pel que s'ha creat el ticket
    UPDATE DW_tickets_per_canal
    SET qtt_tickets_canal = qtt_tickets_canal-1
    WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY') AND canal = canal_in;
END DW_tickets_per_canal_DEC;

CREATE OR REPLACE PROCEDURE DW_tickets_per_canal_MOD (pais_in IN VARCHAR2, canal_antic_in IN VARCHAR2, canal_nou_in IN VARCHAR2)
BEGIN
    -- -1 a qtt_tickets_canal per aquell pais/any/mes i canal_antic
    UPDATE DW_tickets_per_canal
    SET qtt_tickets_canal = qtt_tickets_canal-1
    WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY') AND canal = canal_antic_in;

    -- +1 a qtt_tickets_canal per aquell pais/any/mes i canal_nou
    UPDATE DW_tickets_per_canal
    SET qtt_tickets_canal = qtt_tickets_canal+1
    WHERE pais = pais_in AND mes = to_char(sysdate, 'MM') AND ani = to_char(sysdate, 'YYYY') AND canal = canal_nou_in;
END DW_tickets_per_canal_MOD;

```

11.6 Procediments de consulta del Data Warehouse

```
CREATE OR REPLACE PROCEDURE DW_temps_incidencies_prio_SEL (mes_in IN INTEGER, ani_in IN INTEGER) AS
prio va_params;
temps va_params;

BEGIN

    SELECT prioritat, temps_total/qtt_incidencies BULK COLLECT INTO prio, temps
    FROM DW_temps_incidencies_prio
    WHERE mes = mes_in AND ani = ani_in;

    FOR indx IN 1.. prio.count
    LOOP
        dbms_output.put_line(prio(indx) || ' : ' || temps(indx));
    END LOOP;

END DW_temps_incidencies_prio_SEL;
```

```
CREATE OR REPLACE PROCEDURE DW_petitions_sobre_temps_SEL (mes_in IN INTEGER, ani_in IN INTEGER, pais_in IN VARCHAR2, RSP OUT INTEGER) AS
BEGIN

    SELECT (qtt_petitions_sobre_temps * 100 / qtt_petitions) INTO RSP
    FROM DW_petitions_sobre_temps
    WHERE mes = mes_in AND ani = ani_in AND pais = pais_in;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := NULL; --No existeix cap entrada per aquell mes/any/pais
        WHEN ZERO_DIVIDE THEN RSP := 0; --La qtt_petitions es 0 per aquell mes/any/pais

END DW_petitions_sobre_temps_SEL;
```

```
CREATE OR REPLACE PROCEDURE DW_usuari_mes_incidencies_SEL (RSP OUT VARCHAR2) AS
BEGIN

    SELECT usuari INTO RSP
    FROM (
        SELECT usuari
        FROM DW_usuari_mes_incidencies
        ORDER BY qtt_incidencies DESC
    )
    WHERE ROWNUM = 1;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := NULL; --No existeix cap entrada a la taula

END DW_usuari_mes_incidencies_SEL;
```

```
CREATE OR REPLACE PROCEDURE DW_top10_petitions_seu_SEL (seu_in IN VARCHAR2) AS
ser va_params;
pet va_params;

BEGIN

    SELECT servei, QTT_PETITIONS BULK COLLECT INTO ser, pet
    FROM (
        SELECT servei, QTT_PETITIONS
        FROM DW_top10_petitions_seu
        WHERE seu = seu_in
        ORDER BY qtt_petitions DESC
    )
    WHERE ROWNUM <= 10;

    FOR indx IN 1..ser.count
    LOOP
        dbms_output.put_line(ser(indx) || ' : ' || pet(indx));
    END LOOP;

END DW_top10_petitions_seu_SEL;
```

```

CREATE OR REPLACE PROCEDURE DW_tickets_resolts_mes_SEL (RSP OUT INTEGER) AS
    any_temp INTEGER;
    mes_temp INTEGER;
    tickets_mes_actual INTEGER;
    tickets_mes_anterior INTEGER;
BEGIN
    --seleccionem la quantitat de ticktes del mes i l'any actuals
    SELECT to_char(sysdate, 'YYYY') INTO any_temp FROM dual;
    SELECT to_char(sysdate, 'MM') INTO mes_temp FROM dual;

    SELECT qtt_peticions INTO tickets_mes_actual
    FROM DW_tickets_resolts_mes
    WHERE ani = any_temp AND mes = mes_temp;

    --seleccionem la quantitat de ticktes del mes i l'any anteriors
    mes_temp := mes_temp-1;

    IF mes_temp = 0 THEN
        mes_temp := 12;
        any_temp := any_temp-1 ;
    END IF;

    SELECT qtt_peticions INTO tickets_mes_anterior
    FROM DW_tickets_resolts_mes
    WHERE ani = any_temp AND mes = mes_temp;

    --calculem el percentatge
    SELECT (tickets_mes_actual*100/tickets_mes_anterior)-100 INTO RSP
    FROM dual;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := NULL; --No existeix cap entrada pel mes actual o anterior
        WHEN ZERO_DIVIDE THEN RSP := 0; --Els tickets_mes_anterior es 0
END DW_tickets_resolts_mes_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_servei_peticions_SEL (RSP OUT VARCHAR2) AS
BEGIN
    SELECT servei INTO RSP
    FROM (
        SELECT servei
        FROM DW_servei_peticions
        ORDER BY qtt_peticions DESC
    )
    WHERE ROWNUM = 1;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := NULL; --No existeix cap entrada a la taula
END DW_servei_peticions_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_incidencies_seu_any_SEL (RSP OUT INTEGER) AS
BEGIN
    SELECT seu INTO RSP
    FROM (
        SELECT seu
        FROM DW_incidencies_seu_any
        WHERE ani = to_char(sysdate, 'YYYY')
        ORDER BY qtt_incidencies DESC
    )
    WHERE ROWNUM = 1;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := 0; --No existeix cap entrada a la taula
END DW_incidencies_seu_any_SEL;

```



```

CREATE OR REPLACE PROCEDURE DW_hores_tickets_L3_SEL (ani_in IN INTEGER, RSP OUT INTEGER) AS
BEGIN

    SELECT qtt_hores INTO RSP
    FROM DW_hores_tickets_L3
    WHERE ani = ani_in

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := 0; --No existeix cap entrada a la taula per aquell any

END DW_hores_tickets_L3_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_incidencies_PM_SEL (RSP OUT INTEGER) AS
BEGIN

    SELECT qtt_incidencies INTO RSP
    FROM DW_incidencies_PM
    WHERE ani = to_char(sysdate, 'YYYY');

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := 0; --No existeix cap entrada a la taula per aquell any

END DW_incidencies_PM_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_sms_no_respostos_SEL (RSP OUT INTEGER) AS
BEGIN

    SELECT (qtt_tickets_no_respostos/qtt_tickets*100) INTO RSP
    FROM DW_sms_no_respostos
    WHERE ani = to_char(sysdate, 'YYYY');

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := 0; --No existeix cap entrada a la taula per aquell any

END DW_sms_no_respostos_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_tickets_tancats_SEL (RSP OUT INTEGER) AS
BEGIN

    SELECT qtt_tancats INTO RSP
    FROM DW_tickets_tancats;

    --Tractament Excepcions
    EXCEPTION
        WHEN NO_DATA_FOUND THEN RSP := 0; --No existeix cap entrada a la taula

END DW_tickets_tancats_SEL;

```

```

CREATE OR REPLACE PROCEDURE DW_tickets_per_canal_SEL (mes_in IN VARCHAR2, ani_in IN VARCHAR2, pais_in IN VARCHAR2) AS
can va_params;
percentatge va_params;

BEGIN

    SELECT canal, qtt_tickets_canal/qtt_tickets_total*100 BULK COLLECT INTO can, percentatge
    FROM DW_tickets_per_canal
    WHERE mes = mes_in AND ani = ani_in and pais = pais_in;

    FOR indx IN 1..can.count
    LOOP
        dbms_output.put_line(can(indx) || ' : ' || percentatge(indx));
    END LOOP;

END DW_tickets_per_canal_SEL;

```

11.6 Volum de dades inicials

```
--País
pais_alta ('Espanya', RSP);
pais_alta ('Italia', RSP);
pais_alta ('Franca', RSP);
pais_alta ('Alemanya', RSP);

--Seu
seu_alta ('S001', 'Avd Catalunya 34', NULL, 'Espanya', RSP);
seu_alta ('S002', 'Carrer Armengol 12', NULL, 'Espanya', RSP);
seu_alta ('S003', 'Via del Corso SN', NULL, 'Italia', RSP);
seu_alta ('S004', 'Rue du temple 2', NULL, 'Franca', RSP);

--Usuari
usuari_alta ('sergi_TEC@corp.com', 'S001', RSP);
usuari_alta ('joan_TEC@corp.com', 'S001', RSP);
usuari_alta ('sandra@corp.com', 'S001', RSP);
usuari_alta ('ramon@corp.com', 'S001', RSP);
usuari_alta ('julia@corp.com', 'S001', RSP);
usuari_alta ('josep@corp.com', 'S001', RSP);
usuari_alta ('angel@corp.com', 'S001', RSP);
usuari_alta ('angelina@corp.com', 'S001', RSP);
usuari_alta ('ester@corp.com', 'S001', RSP);
usuari_alta ('alba_TEC@corp.com', 'S001', RSP);
usuari_alta ('ingrid_TEC@corp.com', 'S001', RSP);
usuari_alta ('josepr@corp.com', 'S001', RSP);
usuari_alta ('anna@corp.com', 'S002', RSP);
usuari_alta ('silvia@corp.com', 'S002', RSP);
usuari_alta ('lucia_TEC@corp.com', 'S002', RSP);
usuari_alta ('albert_TEC@corp.com', 'S002', RSP);
usuari_alta ('carmeg@corp.com', 'S002', RSP);
usuari_alta ('ramona@corp.com', 'S002', RSP);
usuari_alta ('blasas@corp.com', 'S002', RSP);
usuari_alta ('ricard_TEC@corp.com', 'S003', RSP);
usuari_alta ('lluis_TEC@corp.com', 'S003', RSP);
usuari_alta ('carla@corp.com', 'S003', RSP);
usuari_alta ('carme@corp.com', 'S003', RSP);

--Interaccio
interaccio_alta ('sandra@corp.com', 'comanda', RSP);
interaccio_alta ('angel@corp.com', 'albara', RSP);
interaccio_alta ('anna@corp.com', 'baixa registre', RSP);
interaccio_alta ('carme@corp.com', 'consulta catalog', RSP);

--NivellSuport
nivellsuport_alta ('L1', 1, RSP);
nivellsuport_alta ('L2', 2, RSP);
nivellsuport_alta ('L3', 3, RSP);
nivellsuport_alta ('Problem Management', 4, RSP);

--GrupSuport
grupsuport_alta ('SUP001', 'Suport L1 1', 'L1', RSP);
grupsuport_alta ('SUP002', 'Suport L2 1', 'L2', RSP);
grupsuport_alta ('SUP003', 'Suport L3 1', 'L3', RSP);
grupsuport_alta ('SUP004', 'Suport PM 1', 'Problem Management', RSP);

--Tecníc
tecnic_alta ('sergi_TEC@corp.com', 'SUP001', RSP);
tecnic_alta ('joan_TEC@corp.com', 'SUP001', RSP);
tecnic_alta ('alba_TEC@corp.com', 'SUP002', RSP);
tecnic_alta ('ingrid_TEC@corp.com', 'SUP002', RSP);
tecnic_alta ('lucia_TEC@corp.com', 'SUP003', RSP);
tecnic_alta ('albert_TEC@corp.com', 'SUP003', RSP);
tecnic_alta ('ricard_TEC@corp.com', 'SUP004', RSP);
tecnic_alta ('lluis_TEC@corp.com', 'SUP004', RSP);

--Prioritat
prioritat_alta (1, RSP);
prioritat_alta (2, RSP);
prioritat_alta (3, RSP);
prioritat_alta (4, RSP);

--GrupServei
grupservei_alta ('Aplicacions', RSP);
grupservei_alta ('Connexions', RSP);
grupservei_alta ('Registres', RSP);
grupservei_alta ('Interfícies', RSP);

--Servei
servei_alta ('SERV001', NULL, 'Aplicacions', 60, 1, RSP);
servei_alta ('SERV002', NULL, 'Aplicacions', 120, 1, RSP);
servei_alta ('SERV003', NULL, 'Connexions', 80, 2, RSP);
servei_alta ('SERV004', NULL, 'Connexions', 30, 3, RSP);
servei_alta ('SERV005', NULL, 'Registres', 60, 3, RSP);
```

```

--OfertaGrupServei
ofertagrupservei_alta ('SUP001', 'SERV001', RSP);
ofertagrupservei_alta ('SUP001', 'SERV002', RSP);
ofertagrupservei_alta ('SUP001', 'SERV003', RSP);
ofertagrupservei_alta ('SUP001', 'SERV004', RSP);
ofertagrupservei_alta ('SUP002', 'SERV001', RSP);
ofertagrupservei_alta ('SUP002', 'SERV002', RSP);
ofertagrupservei_alta ('SUP002', 'SERV004', RSP);
ofertagrupservei_alta ('SUP002', 'SERV005', RSP);
ofertagrupservei_alta ('SUP003', 'SERV001', RSP);
ofertagrupservei_alta ('SUP003', 'SERV002', RSP);
ofertagrupservei_alta ('SUP003', 'SERV003', RSP);
ofertagrupservei_alta ('SUP003', 'SERV004', RSP);
ofertagrupservei_alta ('SUP003', 'SERV005', RSP);
ofertagrupservei_alta ('SUP004', 'SERV001', RSP);
ofertagrupservei_alta ('SUP004', 'SERV002', RSP);
ofertagrupservei_alta ('SUP004', 'SERV003', RSP);
ofertagrupservei_alta ('SUP004', 'SERV004', RSP);
ofertagrupservei_alta ('SUP004', 'SERV005', RSP);

--TipusTicket
tipusticket_alta ('Incidencia', RSP);
tipusticket_alta ('Peticio', RSP);

--Canal
canal_alta ('Web', RSP);
canal_alta ('Xat', RSP);
canal_alta ('Telefon', RSP);

--CreacioTicket
creacioticket_alta ('Agent', RSP);
creacioticket_alta ('Automatic', RSP);

--Ticket
ticket_alta ('T001', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T002', 'SERV003', 2, 'Peticio', 'Xat', 'Agent', 'josepr@corp.com', RSP);
ticket_alta ('T003', 'SERV004', 2, 'Incidencia', 'Web', 'Automatic', 'ramona@corp.com', RSP);
ticket_alta ('T004', 'SERV005', 4, 'Peticio', 'Telefon', 'Automatic', 'carne@corp.com', RSP);

--AssigTicketGrup
assigticketgrup_alta ('T001', 'SUP001', RSP);
assigticketgrup_alta ('T002', 'SUP001', RSP);
assigticketgrup_alta ('T003', 'SUP001', RSP);
assigticketgrup_alta ('T004', 'SUP002', RSP);

--EstatTicket
estatticket_alta ('Creat', RSP);
estatticket_alta ('Assignat', RSP);
estatticket_alta ('En progres', RSP);
estatticket_alta ('Resolt', RSP);
estatticket_alta ('Cancelat', RSP);

--CanviEstat
canviestat_alta ('T001', 'sergi_TEC@corp.com', 'Assignat', RSP);
canviestat_alta ('T002', 'sergi_TEC@corp.com', 'Assignat', RSP);
canviestat_alta ('T003', 'joan_TEC@corp.com', 'Assignat', RSP);
canviestat_alta ('T004', 'alba_TEC@corp.com', 'Assignat', RSP);

--EstatSms
estatsms_alta ('Enviat', RSP);
estatsms_alta ('Rebut', RSP);
estatsms_alta ('Llegit', RSP);

--Missatge
missatge_alta ('SMS001', 'SYS', 'Enviat', 'T001', RSP);
missatge_alta ('SMS002', 'SYS', 'Enviat', 'T002', RSP);
missatge_alta ('SMS003', 'SYS', 'Enviat', 'T003', RSP);
missatge_alta ('SMS004', 'SYS', 'Enviat', 'T004', RSP);
missatge_alta ('SMS005', 'carne@corp.com', 'Enviat', 'T004', RSP);

```

11.7 Jocs de proves

11.7.1 Base de dades operativa

```
--Pais
pais_alta ('Grecia', RSP);
pais_alta ('Alemania', RSP);
pais_baixa ('Franca', RSP);
pais_baixa ('Grecia', RSP);

--Seu
seu_alta ('S005', 'Ronda Gran 3', NULL, 'Espanya', RSP);
seu_alta ('S002', 'Carrer Armengol 12', NULL, 'Espanya', RSP);
seu_alta ('S006', NULL, NULL, 'Espanya', RSP);
seu_alta ('S006', 'Carrer Armengol 12', NULL, 'Portugal', RSP);
seu_baixa ('S001', RSP);
seu_baixa ('S005', RSP);

--Usuari
usuari_alta ('jaume@corp.com', 'S001', RSP);
usuari_alta ('angel@corp.com', 'S001', RSP);
usuari_alta ('john@corp.com', NULL, RSP);
usuari_alta ('john@corp.com', 'S020', RSP);
usuari_modif_seu ('angel@corp.com', NULL, RSP);
usuari_modif_seu ('angel@corp.com', 'S020', RSP);
usuari_modif_seu ('jaume@corp.com', 'S002', RSP);
usuari_baixa ('jaume@corp.com', RSP);
usuari_alta ('jaume@corp.com', 'S001', RSP);
usuari_baixa ('jaume@corp.com', RSP);

--NivellSuport
nivellsuport_alta ('L4', 5, RSP);
nivellsuport_alta ('L2', 2, RSP);
nivellsuport_baixa ('L1', RSP);
nivellsuport_baixa ('L4', RSP);

--GrupSuport
grupsuport_alta ('SUP005', 'Suport L1 2', 'L1', RSP);
grupsuport_alta ('SUP002', 'Suport L2 1', 'L2', RSP);
grupsuport_alta ('SUP006', 'Suport L1 3', NULL, RSP);
grupsuport_alta ('SUP006', 'Suport L1 3', 'L5', RSP);
grupsuport_baixa ('SUP001', RSP);
grupsuport_modif_nom ('SUP002', NULL, RSP);
grupsuport_modif_nom ('SUP005', 'Suport L1 PRO', RSP);
grupsuport_baixa ('SUP005', RSP);

--Tecnica
usuari_alta ('john_TEC@corp.com', 'S001', RSP);
-----
tecnic_alta ('john_TEC@corp.com', 'SUP001', RSP);
tecnic_alta ('joan_TEC@corp.com', 'SUP001', RSP);
tecnic_alta ('jim_TEC@corp.com', NULL, RSP);
tecnic_alta ('jim_TEC@corp.com', 'SUP020', RSP);
tecnic_baixa ('joan_TEC@corp.com', RSP);
tecnic_modif_grup ('joan_TEC@corp.com', 'SUP020', RSP);
tecnic_modif_grup ('john_TEC@corp.com', 'SUP002', RSP);
tecnic_baixa ('john_TEC@corp.com', RSP);
-----
usuari_baixa ('john_TEC@corp.com', RSP);

--Prioritat
prioritat_alta (5, RSP);
prioritat_alta (2, RSP);
prioritat_baixa (1, RSP);
prioritat_baixa (5, RSP);

--GrupServei
grupservei_alta ('Infraestructura', RSP);
grupservei_alta ('Connexions', RSP);
grupservei_baixa ('Infraestructura', RSP);

--Servei
servei_alta ('SERV006', NULL, 'Aplicacions', 120, 3, RSP);
servei_alta ('SERV002', NULL, 'Aplicacions', 120, 1, RSP);
servei_alta ('SERV007', NULL, NULL, 120, 3, RSP);
servei_baixa ('SERV001', RSP);
servei_modif_grup ('SERV006', 'GrupInexistent', RSP);
servei_modif_temps ('SERV006', NULL, RSP);
servei_modif_prioritat ('SERV006', 8, RSP);
servei_modif_descripcio ('SERV006', 'Descripció', RSP);
servei_modif_grup ('SERV006', 'Connexions', RSP);
servei_modif_prioritat ('SERV006', 1, RSP);
servei_modif_temps ('SERV006', 30, RSP);
servei_baixa ('SERV006', RSP);

--OfertaGrupServei
ofertagrupservei_alta ('SUP001', 'SERV005', RSP);
ofertagrupservei_alta ('SUP001', 'SERV001', RSP);
ofertagrupservei_alta ('SUP001', NULL, RSP);
ofertagrupservei_baixa ('SUP001', 'SERV005', RSP);
```

```

--TipusTicket
tipusticket_alta ('Solicitud', RSP);
tipusticket_alta ('Peticio', RSP);
tipusticket_baixa ('Peticio', RSP);
tipusticket_baixa ('Solicitud', RSP);

--Canal
canal_alta ('Missatge', RSP);
canal_alta ('Web', RSP);
canal_baixa ('Xat', RSP);
canal_baixa ('Missatge', RSP);

--CreacioTicket
creacioticket_alta ('Formulari', RSP);
creacioticket_alta ('Automatic', RSP);
creacioticket_baixa ('Automatic', RSP);
creacioticket_baixa ('Formulari', RSP);

--Ticket
ticket_alta ('T005', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T001', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T006', 'SERV001', 1, 'Incidencia', NULL, 'Agent', NULL, RSP);
ticket_alta ('T006', 'SERV020', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T006', 'SERV001', 20, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T006', 'SERV001', 1, 'Solicitud', 'Xat', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T006', 'SERV001', 1, 'Incidencia', 'Carta', 'Agent', 'sandra@corp.com', RSP);
ticket_alta ('T006', 'SERV001', 1, 'Incidencia', 'Xat', 'Formulari', 'sandra@corp.com', RSP);
ticket_baixa ('T001', RSP);
ticket_modif_canal ('T005', 'Carta');
ticket_modif_creacio ('T005', 'Formulari');
ticket_modif_canal ('T005', 'Web');
ticket_modif_creacio ('T005', 'Automatic');
ticket_baixa ('T005', RSP);

--AssigTicketGrup
grupsuport_alta ('SUP005', 'Suport L2 2', 'L2', RSP);
ofertagrupservei_alta ('SUP005', 'SERV003', RSP);
ticket_alta ('T005', 'SERV003', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
-----
assigticketgrup_alta ('T005', 'SUP005', RSP);
assigticketgrup_alta ('T001', 'SUP001', RSP);
assigticketgrup_alta ('T005', 'SUP001', RSP);
assigticketgrup_alta ('T005', 'SUP002', RSP);
assigticketgrup_baixa ('T005', 'SUP005', RSP);
-----
ticket_baixa ('T005', RSP);
ofertagrupservei_baixa ('SUP005', 'SERV003', RSP);
grupsuport_baixa ('SUP005', RSP);

--EstatTicket
estatticket_alta ('Registret', RSP);
estatticket_alta ('Creat', RSP);
estatticket_baixa ('Assignat', RSP);
estatticket_baixa ('Registret', RSP);

--CanviEstat
canviestat_alta ('T001', 'sergi_TEC@corp.com', 'En progres', RSP);
canviestat_alta ('T001', 'alba_TEC@corp.com', 'Resolt', RSP);

--EstatSms
estatsms_alta ('Perdut', RSP);
estatsms_alta ('Rebut', RSP);
estatsms_baixa ('Enviat', RSP);
estatsms_baixa ('Perdut', RSP);

--Missatge
missatge_alta ('SMS006', 'SYS', 'Enviat', 'T001', RSP);
missatge_alta ('SMS002', 'SYS', 'Enviat', 'T002', RSP);
missatge_alta ('SMS007', NULL, 'Enviat', 'T001', RSP);
missatge_modif_estat ('SMS006', 'Perdut', RSP);
missatge_modif_estat ('SMS006', 'Rebut', RSP);
missatge_baixa ('SMS006', RSP);

```

11.7.2 Data Warehouse

DW_temps_incidencies_prio

```
BEGIN
canviestat_alta ('T001', 'sergi_TEC@corp.com', 'Resolt', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');

canviestat_alta ('T003', 'joan_TEC@corp.com', 'Resolt', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');

ticket_alta ('T005', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T005', 'SUP001', RSP);
canviestat_alta ('T005', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');

dbms_lock.sleep(5);
canviestat_alta ('T005', 'sergi_TEC@corp.com', 'Resolt', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');

ticket_alta ('T006', 'SERV001', 4, 'Incidencia', 'Xat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T006', 'SUP001', RSP);
canviestat_alta ('T006', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');
dbms_lock.sleep(1);
canviestat_alta ('T006', 'sergi_TEC@corp.com', 'Resolt', RSP);
DW_temps_incidencies_prio_SEL(05,2018);
dbms_output.put_line('-----');

END;
```

DW_peticions_sobre_temps

```
BEGIN
canviestat_alta ('T002', 'sergi_TEC@corp.com', 'Resolt', RSP);
DW_peticions_sobre_temps_SEL(05,2018,'Espanya', RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T007', 'SERV002', 1, 'Peticio', 'Xat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T007', 'SUP001', RSP);
canviestat_alta ('T007', 'sergi_TEC@corp.com', 'Resolt', RSP);
DW_peticions_sobre_temps_SEL(05,2018,'Espanya', RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

END;
```

DW_usuari_mes_incidencies

```
BEGIN
ticket_alta ('T008', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T008', 'SUP001', RSP);
canviestat_alta ('T008', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T009', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T009', 'SUP001', RSP);
canviestat_alta ('T009', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T010', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T010', 'SUP001', RSP);
canviestat_alta ('T010', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T011', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T011', 'SUP001', RSP);
canviestat_alta ('T011', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T012', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T012', 'SUP001', RSP);
canviestat_alta ('T012', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T013', 'SERV001', 1, 'Incidencia', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T013', 'SUP001', RSP);
canviestat_alta ('T003', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_usuari_mes_incidencies_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

END;
```

DW_top10_peticions_seu

□ BEGIN

```

ticket_alta ('T014', 'SERV004', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T014', 'SUP001', RSP);
canviestat_alta ('T014', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T015', 'SERV004', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T015', 'SUP001', RSP);
canviestat_alta ('T015', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T016', 'SERV004', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T016', 'SUP001', RSP);
canviestat_alta ('T016', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T017', 'SERV004', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T017', 'SUP001', RSP);
canviestat_alta ('T017', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_top10_peticions_seu_SEL('S001');
dbms_output.put_line('-----');

ticket_alta ('T018', 'SERV001', 1, 'Peticio', 'Mat', 'Agent', 'anna@corp.com', RSP);
assigticketgrup_alta ('T018', 'SUP001', RSP);
canviestat_alta ('T018', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T019', 'SERV001', 1, 'Peticio', 'Mat', 'Agent', 'anna@corp.com', RSP);
assigticketgrup_alta ('T019', 'SUP001', RSP);
canviestat_alta ('T019', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T020', 'SERV001', 1, 'Peticio', 'Mat', 'Agent', 'anna@corp.com', RSP);
assigticketgrup_alta ('T020', 'SUP001', RSP);
canviestat_alta ('T020', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T021', 'SERV001', 1, 'Peticio', 'Mat', 'Agent', 'anna@corp.com', RSP);
assigticketgrup_alta ('T021', 'SUP001', RSP);
canviestat_alta ('T021', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T022', 'SERV001', 1, 'Peticio', 'Mat', 'Agent', 'anna@corp.com', RSP);
assigticketgrup_alta ('T022', 'SUP001', RSP);
canviestat_alta ('T022', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_top10_peticions_seu_SEL('S002');
dbms_output.put_line('-----');

ticket_alta ('T023', 'SERV003', 1, 'Peticio', 'Mat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T023', 'SUP001', RSP);
canviestat_alta ('T023', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T024', 'SERV003', 1, 'Peticio', 'Mat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T024', 'SUP001', RSP);
canviestat_alta ('T024', 'sergi_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T025', 'SERV003', 1, 'Peticio', 'Mat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T025', 'SUP001', RSP);
canviestat_alta ('T025', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_top10_peticions_seu_SEL('S003');
dbms_output.put_line('-----');

ticket_alta ('T026', 'SERV005', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T026', 'SUP002', RSP);
canviestat_alta ('T026', 'alba_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T027', 'SERV005', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T027', 'SUP002', RSP);
canviestat_alta ('T027', 'alba_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T028', 'SERV005', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T028', 'SUP002', RSP);
canviestat_alta ('T028', 'alba_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T029', 'SERV005', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T029', 'SUP002', RSP);
canviestat_alta ('T029', 'alba_TEC@corp.com', 'Assignat', RSP);
DW_top10_peticions_seu_SEL('S001');
dbms_output.put_line('-----');

ticket_alta ('T030', 'SERV003', 1, 'Peticio', 'Mat', 'Agent', 'sandra@corp.com', RSP);
assigticketgrup_alta ('T030', 'SUP001', RSP);
canviestat_alta ('T030', 'sergi_TEC@corp.com', 'Assignat', RSP);
DW_top10_peticions_seu_SEL('S001');
dbms_output.put_line('-----');

```

END;

DW_tickets_resolts_mes

```
BEGIN

INSERT INTO DW_tickets_resolts_mes VALUES (to_char(sysdate, 'MM')-1, to_char(sysdate, 'YYYY'), 3);
DW_tickets_resolts_mes_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

UPDATE DW_tickets_resolts_mes SET qtt_peticions = 18 WHERE mes = to_char(sysdate, 'MM')-1 AND ani = to_char(sysdate, 'YYYY');
DW_tickets_resolts_mes_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

END;
```

DW_servei_peticions

```
BEGIN

ticket_alta ('T130', 'SERV005', 1, 'Peticio', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T130', 'SUP002', RSP);
canviestat_alta ('T130', 'alba_TEC@corp.com', 'Assignat', RSP);
DW_servei_peticions_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

canviestat_alta ('T026', 'alba_TEC@corp.com', 'Cancelat', RSP);
DW_servei_peticions_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

ticket_alta ('T031', 'SERV004', 1, 'Peticio', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T031', 'SUP002', RSP);
canviestat_alta ('T031', 'alba_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T032', 'SERV004', 1, 'Peticio', 'Xat', 'Agent', 'angel@corp.com', RSP);
assigticketgrup_alta ('T032', 'SUP002', RSP);
canviestat_alta ('T032', 'alba_TEC@corp.com', 'Assignat', RSP);
DW_servei_peticions_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

END;
```

DW_incidencies_seu_any

```
BEGIN

ticket_alta ('T033', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T033', 'SUP002', RSP);
canviestat_alta ('T033', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T034', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T034', 'SUP002', RSP);
canviestat_alta ('T034', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T035', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T035', 'SUP002', RSP);
canviestat_alta ('T035', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T036', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T036', 'SUP002', RSP);
canviestat_alta ('T036', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T037', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T037', 'SUP002', RSP);
canviestat_alta ('T037', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T038', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T038', 'SUP002', RSP);
canviestat_alta ('T038', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T039', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T039', 'SUP002', RSP);
canviestat_alta ('T039', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T040', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T040', 'SUP002', RSP);
canviestat_alta ('T040', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T041', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T041', 'SUP002', RSP);
canviestat_alta ('T041', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T042', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T042', 'SUP002', RSP);
canviestat_alta ('T042', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T043', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T043', 'SUP002', RSP);
canviestat_alta ('T043', 'alba_TEC@corp.com', 'Assignat', RSP);

ticket_alta ('T044', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'carla@corp.com', RSP);
assigticketgrup_alta ('T044', 'SUP002', RSP);
canviestat_alta ('T044', 'alba_TEC@corp.com', 'Assignat', RSP);

DW_incidencies_seu_any_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

END;
```


DW_hores_tickets_L3

```
BEGIN

    ticket_alta ('T045', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'ramona@corp.com', RSP);
    assigticketgrup_alta ('T045', 'SUP003', RSP);
    canviestat_alta ('T045', 'lucia_TEC@corp.com', 'Assignat', RSP);
    ticket_alta ('T046', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'ramona@corp.com', RSP);
    assigticketgrup_alta ('T046', 'SUP003', RSP);
    canviestat_alta ('T046', 'lucia_TEC@corp.com', 'Assignat', RSP);
    DW_hores_tickets_L3_SEL(2018, RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

    /*ESPERA D'UNES QUANTES HORES*/

    assigticketgrup_alta ('T045', 'SUP004', RSP);
    canviestat_alta ('T045', 'ricard_TEC@corp.com', 'Assignat', RSP);
    DW_hores_tickets_L3_SEL(2018, RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

    canviestat_alta ('T046', 'lucia_TEC@corp.com', 'Resolt', RSP);
    DW_hores_tickets_L3_SEL(2018, RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

END;
```

DW_incidencies_PM

```
BEGIN

    ticket_alta ('T047', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'ramona@corp.com', RSP);
    assigticketgrup_alta ('T047', 'SUP003', RSP);
    canviestat_alta ('T047', 'lucia_TEC@corp.com', 'Assignat', RSP);
    dbms_lock.sleep(1);
    assigticketgrup_alta ('T047', 'SUP004', RSP);
    canviestat_alta ('T047', 'ricard_TEC@corp.com', 'Assignat', RSP);

    ticket_alta ('T048', 'SERV005', 1, 'Incidencia', 'Xat', 'Agent', 'ramona@corp.com', RSP);
    assigticketgrup_alta ('T048', 'SUP003', RSP);
    canviestat_alta ('T048', 'lucia_TEC@corp.com', 'Assignat', RSP);
    dbms_lock.sleep(1);
    assigticketgrup_alta ('T048', 'SUP004', RSP);
    canviestat_alta ('T048', 'ricard_TEC@corp.com', 'Assignat', RSP);

    DW_incidencies_PM_SEL(RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

    assigticketgrup_baixa ('T048', 'SUP004', RSP);

    DW_incidencies_PM_SEL(RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

END;
```

DW_sms_no_respostos

```
BEGIN

    DW_sms_no_respostos_SEL(RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

    missatge_alta ('SMS007', 'sandra@corp.com', 'Enviat', 'T001', RSP);
    missatge_alta ('SMS008', 'josepr@corp.com', 'Enviat', 'T002', RSP);
    missatge_alta ('SMS009', 'ramona@corp.com', 'Enviat', 'T003', RSP);
    missatge_alta ('SMS010', 'sandra@corp.com', 'Enviat', 'T007', RSP);
    missatge_alta ('SMS011', 'angel@corp.com', 'Enviat', 'T008', RSP);
    missatge_alta ('SMS012', 'angel@corp.com', 'Enviat', 'T009', RSP);
    missatge_alta ('SMS013', 'angel@corp.com', 'Enviat', 'T010', RSP);
    missatge_alta ('SMS014', 'angel@corp.com', 'Enviat', 'T011', RSP);
    missatge_alta ('SMS015', 'angel@corp.com', 'Enviat', 'T012', RSP);
    missatge_alta ('SMS016', 'angel@corp.com', 'Enviat', 'T013', RSP);

    DW_sms_no_respostos_SEL(RSP);
    DBMS_OUTPUT.put_line(RSP);
    dbms_output.put_line('-----');

END;
```

DW_tickets_tancats

BEGIN

```
canviestat_alta ('T033', 'alba_TEC@corp.com', 'Resolt', RSP);
canviestat_alta ('T034', 'alba_TEC@corp.com', 'Resolt', RSP);
canviestat_alta ('T035', 'alba_TEC@corp.com', 'Resolt', RSP);
DW_tickets_tancats_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');

canviestat_alta ('T036', 'alba_TEC@corp.com', 'Cancelat', RSP);
canviestat_alta ('T037', 'alba_TEC@corp.com', 'Cancelat', RSP);
DW_tickets_tancats_SEL(RSP);
DBMS_OUTPUT.put_line(RSP);
dbms_output.put_line('-----');
```

END;

DW_tickets_per_canal

BEGIN

```
ticket_alta ('T049', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'julia@corp.com', RSP);
assigticketgrup_alta ('T049', 'SUP003', RSP);
canviestat_alta ('T049', 'lucia_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T050', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'julia@corp.com', RSP);
assigticketgrup_alta ('T050', 'SUP003', RSP);
canviestat_alta ('T050', 'lucia_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T051', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'julia@corp.com', RSP);
assigticketgrup_alta ('T051', 'SUP003', RSP);
canviestat_alta ('T051', 'lucia_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T052', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'julia@corp.com', RSP);
assigticketgrup_alta ('T052', 'SUP003', RSP);
canviestat_alta ('T052', 'lucia_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T053', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'julia@corp.com', RSP);
assigticketgrup_alta ('T053', 'SUP003', RSP);
canviestat_alta ('T053', 'lucia_TEC@corp.com', 'Assignat', RSP);
DW_tickets_per_canal_SEL(05, 2018, 'Espanya');
dbms_output.put_line('-----');

ticket_alta ('T054', 'SERV005', 1, 'Incidencia', 'Telefon', 'Agent', 'carme@corp.com', RSP);
assigticketgrup_alta ('T054', 'SUP003', RSP);
canviestat_alta ('T054', 'lucia_TEC@corp.com', 'Assignat', RSP);
ticket_alta ('T055', 'SERV005', 1, 'Incidencia', 'Telefon', 'Agent', 'carme@corp.com', RSP);
assigticketgrup_alta ('T055', 'SUP003', RSP);
canviestat_alta ('T055', 'lucia_TEC@corp.com', 'Assignat', RSP);
DW_tickets_per_canal_SEL(05, 2018, 'Italia');
dbms_output.put_line('-----');

usuari_alta ('jan@corp.com', 'S004', RSP);
ticket_alta ('T056', 'SERV005', 1, 'Incidencia', 'Web', 'Agent', 'jan@corp.com', RSP);
assigticketgrup_alta ('T056', 'SUP003', RSP);
canviestat_alta ('T056', 'lucia_TEC@corp.com', 'Assignat', RSP);
DW_tickets_per_canal_SEL(05, 2018, 'Franca');
dbms_output.put_line('-----');
```

END;

11.8 Execució dels jocs i resultats

11.8.1 Base de dades operativa

➤ País

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
127	27/04/18 12:32:05,473000000	pais_alta	OK	SYSTEM.VA_PARAMS('Grecia')
128	27/04/18 12:32:05,473000000	pais_alta	ERROR: Violacio de clau primaria. El pais ja existeix	SYSTEM.VA_PARAMS('Alemania')
129	27/04/18 12:32:05,473000000	pais_baixa	ERROR: Violacio clau forana. Hi ha alguna seu ubicada en aquest pais	SYSTEM.VA_PARAMS('Franca')
130	27/04/18 12:32:05,474000000	pais_baixa	OK	SYSTEM.VA_PARAMS('Grecia')

➤ Seu

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
131	27/04/18 12:36:08,868000000	seu_alta	OK	SYSTEM.VA_PARAMS('S005', 'Ronda Gran 3', null, 'Espanya')
132	27/04/18 12:36:08,868000000	seu_alta	ERROR: Violacio de clau primaria. La seu ja existeix	SYSTEM.VA_PARAMS('S002', 'Carrer Armengol 12', null, 'Espanya')
133	27/04/18 12:36:08,868000000	seu_alta	ERROR: Null. A excepcio de la tipologia, tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('S006', null, null, 'Espanya')
134	27/04/18 12:36:08,869000000	seu_alta	ERROR: Violacio de clau forana. El pais no existeix	SYSTEM.VA_PARAMS('S006', 'Carrer Armengol 12', null, 'Portugal')
135	27/04/18 12:36:08,871000000	seu_baixa	ERROR: Violacio clau forana. Existeixen usuaris inscrits en aquesta seu	SYSTEM.VA_PARAMS('S001')
136	27/04/18 12:36:08,871000000	seu_baixa	OK	SYSTEM.VA_PARAMS('S005')

➤ Usuari

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
466	30/04/18 09:29:17,947000000	usuari_alta	OK	SYSTEM.VA_PARAMS('jaume@corp.com', 'S001')
467	30/04/18 09:29:17,947000000	usuari_alta	ERROR: Violacio de clau primaria. Aques usuari ja existeix	SYSTEM.VA_PARAMS('angel@corp.com', 'S001')
468	30/04/18 09:29:17,947000000	usuari_alta	ERROR: Null. A excepcio de la data de baixa, tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('john@corp.com', null)
469	30/04/18 09:29:17,948000000	usuari_alta	ERROR: Violacio de clau forana. La seu no existeix	SYSTEM.VA_PARAMS('john@corp.com', 'S020')
470	30/04/18 09:29:17,948000000	usuari_modif_seu	ERROR: Null. La seu no pot ser nula	SYSTEM.VA_PARAMS('angel@corp.com', null)
471	30/04/18 09:29:17,948000000	usuari_modif_seu	ERROR: Violacio de clau forana. La seu no existeix	SYSTEM.VA_PARAMS('angel@corp.com', 'S020')
472	30/04/18 09:29:17,948000000	usuari_modif_seu	OK	SYSTEM.VA_PARAMS('jaume@corp.com', 'S002')
473	30/04/18 09:29:17,948000000	usuari_baixa	OK	SYSTEM.VA_PARAMS('jaume@corp.com')
474	30/04/18 09:29:17,948000000	usuari_alta	OK	SYSTEM.VA_PARAMS('jaume@corp.com', 'S001')

➤ NivellSuport

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
119	27/04/18 12:58:45,002000000	nivellsuport_alta	OK	SYSTEM.VA_PARAMS('L4')
120	27/04/18 12:58:45,002000000	nivellsuport_alta	ERROR: Violacio de clau primaria. El nivell ja existeix	SYSTEM.VA_PARAMS('L2')
121	27/04/18 12:58:45,002000000	nivellsuport_baixa	ERROR: Violacio clau forana. Hi ha alguna grup de suport al que correspon aquest nivell	SYSTEM.VA_PARAMS('L1')
122	27/04/18 12:58:45,003000000	nivellsuport_baixa	OK	SYSTEM.VA_PARAMS('L4')

➤ GrupSuport

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
123	27/04/18 13:02:40,700000000	grupsuport_alta	OK	SYSTEM.VA_PARAMS('SUP005', 'Suport L1 2', 'L1')
124	27/04/18 13:02:40,701000000	grupsuport_alta	ERROR: Violacio de clau primaria. Aques grup ja existeix	SYSTEM.VA_PARAMS('SUP002', 'Suport L2 1', 'L2')
125	27/04/18 13:02:40,701000000	grupsuport_alta	ERROR: Null. Tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('SUP006', 'Suport L1 3', null)
126	27/04/18 13:02:40,701000000	grupsuport_alta	ERROR: Violacio de clau forana. El nivell no existeix	SYSTEM.VA_PARAMS('SUP006', 'Suport L1 3', 'L5')
127	27/04/18 13:02:40,704000000	grupsuport_baixa	ERROR: Violacio clau forana. Aquest grup esta relacionat amb algun ticket o servei	SYSTEM.VA_PARAMS('SUP001')
128	27/04/18 13:02:40,705000000	grupsuport_modi...	ERROR: Null. El nom no pot ser nul	SYSTEM.VA_PARAMS('SUP002', null)
129	27/04/18 13:02:40,706000000	grupsuport_modi...	OK	SYSTEM.VA_PARAMS('SUP005', 'Suport L1 PRO')
130	27/04/18 13:02:40,707000000	grupsuport_baixa	OK	SYSTEM.VA_PARAMS('SUP005')

➤ Tècnic

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
399	30/04/18 09:16:07,....	usuari_alta	OK	SYSTEM.VA_PARAMS('john_TEC@corp.com', 'S001')
400	30/04/18 09:16:07,....	tecnic_alta	OK	SYSTEM.VA_PARAMS('john_TEC@corp.com', 'SUP001')
401	30/04/18 09:16:07,....	tecnic_alta	ERROR: Violacio de clau primaria. Aques tecnic ja existeix	SYSTEM.VA_PARAMS('joan_TEC@corp.com', 'SUP001')
402	30/04/18 09:16:07,....	tecnic_alta	ERROR: Null. Tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('jim_TEC@corp.com', null)
403	30/04/18 09:16:07,....	tecnic_alta	ERROR: Violacio de clau forana. Elgrup no existeix o el tecnic no es cap usuari	SYSTEM.VA_PARAMS('jim_TEC@corp.com', 'SUP020')
404	30/04/18 09:16:07,....	tecnic_baixa	ERROR: Violacio clau forana. Aquest tecnic esta relacionat amb algun canvi d'estat d'un ticket	SYSTEM.VA_PARAMS('joan_TEC@corp.com')
405	30/04/18 09:16:07,....	tecnic_modif_grup	ERROR: Violacio de clau forana. El grup no existeix	SYSTEM.VA_PARAMS('joan_TEC@corp.com', 'SUP020')
406	30/04/18 09:16:07,....	tecnic_modif_grup	OK	SYSTEM.VA_PARAMS('john_TEC@corp.com', 'SUP002')
407	30/04/18 09:16:07,....	tecnic_baixa	OK	SYSTEM.VA_PARAMS('john_TEC@corp.com')
408	30/04/18 09:16:07,....	usuari_baixa	OK	SYSTEM.VA_PARAMS('john_TEC@corp.com')

➤ Prioritat

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
149	27/04/18 13:18:14,265000000	prioritat_alta	OK	SYSTEM.VA_PARAMS('5')
150	27/04/18 13:18:14,266000000	prioritat_alta	ERROR: Violacio de clau primaria. La prioritat ja existeix	SYSTEM.VA_PARAMS('2')
151	27/04/18 13:18:14,268000000	prioritat_baixa	ERROR: Violacio clau forana. Hi ha algun ticket o servei amb aquesta prioritat	SYSTEM.VA_PARAMS('1')
152	27/04/18 13:18:14,269000000	prioritat_baixa	OK	SYSTEM.VA_PARAMS('5')

➤ GrupServei

CODI	DATA_HORA	PROCEDIMENT	RSP	PARAM_IN
159	27/04/18 13:19:50,896000000	grupservei_alta	OK	SYSTEM.VA_PARAMS('Infraestructura')
160	27/04/18 13:19:50,897000000	grupservei_alta	ERROR: Violacio de clau primaria. El grup ja existeix	SYSTEM.VA_PARAMS('Connexions')
161	27/04/18 13:19:50,897000000	grupservei_baixa	OK	SYSTEM.VA_PARAMS('Infraestructura')

➤ Servei

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
162	27/04/18 13:28:33,202000000	servei_alta	OK	SYSTEM.VA_PARAMS('SERV006', null, 'Aplicacions', '120', '3')
163	27/04/18 13:28:33,202000000	servei_alta	ERROR: Violacio de clau primaria. Aques servei ja existeix	SYSTEM.VA_PARAMS('SERV002', null, 'Aplicacions', '120', '1')
164	27/04/18 13:28:33,202000000	servei_alta	ERROR: Null. A excepcio de la descripcio, tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('SERV007', null, null, '120', '3')
165	27/04/18 13:28:33,204000000	servei_baixa	ERROR: Violacio clau forana. Aquest servei esta relacionat amb algun ticket	SYSTEM.VA_PARAMS('SERV001')
166	27/04/18 13:28:33,205000000	servei_modif_grup	ERROR: Violacio de clau forana. El grup no existeix	SYSTEM.VA_PARAMS('SERV006', 'GrupInexistent')
167	27/04/18 13:28:33,206000000	servei_modif_temps	ERROR: Null. El temps mig no pot ser nul	SYSTEM.VA_PARAMS('SERV006', null)
168	27/04/18 13:28:33,207000000	servei_modif_pr...	ERROR: Violacio de clau forana. LA prioritat no existeix	SYSTEM.VA_PARAMS('SERV006', '8')
169	27/04/18 13:28:33,208000000	servei_modif_de...	OK	SYSTEM.VA_PARAMS('SERV006', 'Descripcio')
170	27/04/18 13:28:33,208000000	servei_modif_grup	OK	SYSTEM.VA_PARAMS('SERV006', 'Connexions')
171	27/04/18 13:28:33,208000000	servei_modif_pr...	OK	SYSTEM.VA_PARAMS('SERV006', '1')
172	27/04/18 13:28:33,208000000	servei_modif_temps	OK	SYSTEM.VA_PARAMS('SERV006', '30')
173	27/04/18 13:28:33,209000000	servei_baixa	OK	SYSTEM.VA_PARAMS('SERV006')

➤ OfertaGrupServei

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
174	27/04/18 13:31:49,180000000	ofertagrupservei_alta	OK	SYSTEM.VA_PARAMS('SUP001', 'SERV005')
175	27/04/18 13:31:49,180000000	ofertagrupservei_alta	ERROR: Violacio de clau primaria. El conjunt (grup - servei) ja existeix	SYSTEM.VA_PARAMS('SUP001', 'SERV001')
176	27/04/18 13:31:49,180000000	ofertagrupservei_alta	ERROR: Null. Tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('SUP001', null)
177	27/04/18 13:31:49,182000000	ofertagrupservei_baixa	OK	SYSTEM.VA_PARAMS('SUP001', 'SERV005')

➤ TipusTicket

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
178	27/04/18 13:35:08,962000000	tipusticket_alta	OK	SYSTEM.VA_PARAMS('Solicitud')
179	27/04/18 13:35:08,962000000	tipusticket_alta	ERROR: Violacio de clau primaria. El tipus ja existeix	SYSTEM.VA_PARAMS('Peticio')
180	27/04/18 13:35:08,964000000	tipusticket_baixa	ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest tipus	SYSTEM.VA_PARAMS('Peticio')
181	27/04/18 13:35:08,964000000	tipusticket_baixa	OK	SYSTEM.VA_PARAMS('Solicitud')

➤ Canal

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
182	27/04/18 13:35:39,645000000	canal_alta	OK	SYSTEM.VA_PARAMS('Missatge')
183	27/04/18 13:35:39,645000000	canal_alta	ERROR: Violacio de clau primaria. El canal ja existeix	SYSTEM.VA_PARAMS('Web')
184	27/04/18 13:35:39,648000000	canal_baixa	ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest canal	SYSTEM.VA_PARAMS('Xat')
185	27/04/18 13:35:39,648000000	canal_baixa	OK	SYSTEM.VA_PARAMS('Missatge')

➤ CrecioTicket

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
186	27/04/18 13:36:07,700000000	creaciotticket_alta	OK	SYSTEM.VA_PARAMS('Formulari')
187	27/04/18 13:36:07,700000000	creaciotticket_alta	ERROR: Violacio de clau primaria. El metode de creacio ja existeix	SYSTEM.VA_PARAMS('Automatic')
188	27/04/18 13:36:07,702000000	creaciotticket_baixa	ERROR: Violacio clau forana. Hi ha algun ticket al que correspon aquest metode de creacio	SYSTEM.VA_PARAMS('Automatic')
189	27/04/18 13:36:07,703000000	creaciotticket_baixa	OK	SYSTEM.VA_PARAMS('Formulari')

➤ Ticket

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
287	27/04/18 14:06:40,116000000	ticket_alta	OK	SYSTEM.VA_PARAMS('T005', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV001', '1',
288	27/04/18 14:06:40,116000000	ticket_alta	ERROR: Violacio de clau primaria. Aques ticket ja existeix	SYSTEM.VA_PARAMS('T001', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV001', '1',
289	27/04/18 14:06:40,116000000	ticket_alta	ERROR: Null. Tots els atributs han de tenir algun valor	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV001', '1',
290	27/04/18 14:06:40,116000000	ticket_alta	ERROR: Violacio de clau forana. El servei, prioritat, tipus, can...	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV020', '1',
291	27/04/18 14:06:40,116000000	ticket_alta	ERROR: Violacio de clau forana. El servei, prioritat, tipus, can...	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV001', '20'
292	27/04/18 14:06:40,116000000	ticket_alta	ERROR: Violacio de clau forana. El servei, prioritat, tipus, can...	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,116000 EUROPE/PARIS', 'SERV001', '1',
293	27/04/18 14:06:40,117000000	ticket_alta	ERROR: Violacio de clau forana. El servei, prioritat, tipus, can...	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,117000 EUROPE/PARIS', 'SERV001', '1',
294	27/04/18 14:06:40,117000000	ticket_alta	ERROR: Violacio de clau forana. El servei, prioritat, tipus, can...	SYSTEM.VA_PARAMS('T006', '27/04/18 14:06:40,117000 EUROPE/PARIS', 'SERV001', '1',
295	27/04/18 14:06:40,117000000	ticket_baixa	ERROR: Violacio clau forana. Aquest ticket esta relacionat amb a...	SYSTEM.VA_PARAMS('T001')
296	27/04/18 14:06:40,117000000	ticket_modif_canal	ERROR: Violacio de clau forana. El canal no existeix	SYSTEM.VA_PARAMS('T005', 'Carta')
297	27/04/18 14:06:40,120000000	ticket_modif_crecio	ERROR: Violacio de clau forana. El metode de creacio no existeix	SYSTEM.VA_PARAMS('T005', 'Formulari')
298	27/04/18 14:06:40,120000000	ticket_modif_canal	OK	SYSTEM.VA_PARAMS('T005', 'Web')
299	27/04/18 14:06:40,120000000	ticket_modif_crecio	OK	SYSTEM.VA_PARAMS('T005', 'Automatic')
300	27/04/18 14:06:40,120000000	ticket_baixa	OK	SYSTEM.VA_PARAMS('T005')

➤ AssigTicketGrup

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
368	30/04/18 08:58:08,...	grupsuport_alta	OK	SYSTEM.VA_PARAMS('SUP005', 'Suport L2 :
369	30/04/18 08:58:08,...	ofertagrupservei_alta	OK	SYSTEM.VA_PARAMS('SUP005', 'SERV003')
370	30/04/18 08:58:08,...	ticket_alta	OK	SYSTEM.VA_PARAMS('T005', '30/04/18 08:5
371	30/04/18 08:58:08,...	assigticketgrup_alta	OK	SYSTEM.VA_PARAMS('T005', 'SUP005')
372	30/04/18 08:58:08,...	assigticketgrup_alta	ERROR: Violacio de clau primaria. Aquesta assignacio ja existeix	SYSTEM.VA_PARAMS('T005', 'SUP005')
373	30/04/18 08:58:08,...	assigticketgrup_alta	ERROR: Restriccio textual. El nivell del grup que s'intenta assignar al ticket es inferior al nivell del grup qu...	SYSTEM.VA_PARAMS('T001', 'SUP001')
374	30/04/18 08:58:08,...	assigticketgrup_alta	ERROR: Restriccio textual. El grup que s'intenta assignar al ticket no ofereix el servei que el ticket demana	SYSTEM.VA_PARAMS('T005', 'SUP002')
375	30/04/18 08:58:08,...	assigticketgrup_baixa	OK	SYSTEM.VA_PARAMS('T005', 'SUP005')
376	30/04/18 08:58:08,...	ticket_baixa	OK	SYSTEM.VA_PARAMS('T005')
377	30/04/18 08:58:08,...	ofertagrupservei_baixa	OK	SYSTEM.VA_PARAMS('SUP005', 'SERV003')
378	30/04/18 08:58:08,...	grupsuport_baixa	OK	SYSTEM.VA_PARAMS('SUP005')

➤ EstatTicket

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
475	30/04/18 09:33:57,596000000	estatticket_alta	OK	SYSTEM.VA_PARAMS('Registrat')
476	30/04/18 09:33:57,596000000	estatticket_alta	ERROR: Violacio de clau primaria. Aques estat ja existeix	SYSTEM.VA_PARAMS('Creat')
477	30/04/18 09:33:57,598000000	estatticket_baixa	ERROR: Violacio clau forana. Aquest estat esta relacionat amb algun ticket	SYSTEM.VA_PARAMS('Assignat')
478	30/04/18 09:33:57,599000000	estatticket_baixa	OK	SYSTEM.VA_PARAMS('Registrat')

➤ CanviEstat

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
490	30/04/18 09:47:24,772000000	canviestat_alta	OK	SYSTEM.VA_PARAMS('T001', 'seergi_TEC@corp.com', 'En progres')
491	30/04/18 09:47:24,772000000	canviestat_alta	ERROR: Restriccio textual. El tecnic no pertany al grup que actualment s'encarrega del ticket	SYSTEM.VA_PARAMS('T001', 'alba_TEC@corp.com', 'Result')

➤ EstatSms

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
492 30/04/18 09:49:50,470000000	estatssms_alta	OK		SYSTEM.VA_PARAMS('Perdut')
493 30/04/18 09:49:50,470000000	estatssms_alta	ERROR: Violacio de clau primaria. Aques estat ja existeix		SYSTEM.VA_PARAMS('Rebut')
494 30/04/18 09:49:50,472000000	estatssms_baixa	ERROR: Violacio clau forana. Aquest estat esta relacionat amb algun missatge		SYSTEM.VA_PARAMS('Enviat')
495 30/04/18 09:49:50,472000000	estatssms_baixa	OK		SYSTEM.VA_PARAMS('Perdut')

➤ Missatge

❖ CODI	❖ DATA_HORA	❖ PROCEDIMENT	❖ RSP	PARAM_IN
496 30/04/18 09:58:31,419000000	missatge_alta	OK		SYSTEM.VA_PARAMS('SMS006', 'SYS', 'Enviat', 'T001')
497 30/04/18 09:58:31,419000000	missatge_alta	ERROR: Violacio de clau primaria. Aques missatge ja existeix		SYSTEM.VA_PARAMS('SMS002', 'SYS', 'Enviat', 'T002')
498 30/04/18 09:58:31,419000000	missatge_alta	ERROR: Null. Tots els atributs han de tenir algun valor		SYSTEM.VA_PARAMS('SMS007', null, 'Enviat', 'T001')
499 30/04/18 09:58:31,421000000	missatge_modif_estat	ERROR: Violacio de clau forana. L'estat no existeix		SYSTEM.VA_PARAMS('SMS006', 'Perdut')
500 30/04/18 09:58:31,421000000	missatge_modif_estat	OK		SYSTEM.VA_PARAMS('SMS006', 'Rebut')

11.8.2 Data warehouse

➤ DW_temps_incidencies_prio

- Canvi d'estat d'incidència de prioritat 1 a resolta

1 : 67

- Canvi d'estat d'incidència de prioritat 2 a resolta

1 : 67

2 : 108

- Creació de nova incidència de prioritat 1 a estat 'Assignat'

1 : 67

2 : 108

- Canvi d'estat de la nova incidència de prioritat 1 a 'Resolta'

1 : 68

2 : 108

- Creació de nova incidència de prioritat 4 a estat 'Assignat'

1 : 68

2 : 108

- Canvi d'estat de la nova incidència de prioritat 4 a 'Resolta'

1 : 68

2 : 108

4 : 26

➤ DW_peticions_sobre_temps

- Canvi d'estat de petició de l'usuari ubicat a Espanya a resolta

100

- Creació d'una altra petició per un usuari ubicat a la seu d'Espanya i posterior canvi d'estat d'aquesta petició de l'usari ubicat a Espanya a resolta (per sota del temps mig de resolució perquè l'acabem de crear).

50

MES	ANI	PAIS	QTT_PETICIONS	QTT_PETICIONS_SOBRE_TEMPS
4	2018	Espanya	2	1

➤ DW_usuari_mes_incidencies

- Creció de la primera incidencia de l'usuri angel
sandra@corp.com
- Creció de la segon incidencia de l'usuri angel
sandra@corp.com
- Creció de la tercera incidencia de l'usuri angel
sandra@corp.com
- Creció de la quarta incidencia de l'usuri angel
angel@corp.com
- Creció de la cinquena incidencia de l'usuri angel
angel@corp.com
- Creció de la sisena incidencia de l'usuri angel
angel@corp.com

USUARI	QTT_INCIDENCIES
sandra@corp.com	3
ramona@corp.com	1
angel@corp.com	6

➤ **DW_top10_peticions_seu**

- Creció de 4 peticions al servei SERV004 de la seu S001

```
SERV004 : 4  
SERV003 : 1  
SERV002 : 1
```

- Creació de 5 peticions al servei SERV001 de la seu S002

```
SERV001 : 5
```

- Creació de 3 peticions al servei SERV003 de la seu S003

```
SERV003 : 3  
SERV005 : 1
```

- Creació de 4 peticions al servei SERV005 de la seu S001

```
SERV004 : 4  
SERV005 : 4  
SERV003 : 1  
SERV002 : 1
```

- Creació de 1 peticions al serveu SERV003 de la seu S001

```
SERV004 : 4  
SERV005 : 4  
SERV003 : 2  
SERV002 : 1
```

SEU	SERVEI	QTT_PETICIONS
S002	SERV001	5
S003	SERV003	3
S001	SERV003	2
S003	SERV005	1
S001	SERV002	1
S001	SERV004	4
S001	SERV005	4

➤ **DW_tickets_resolts_mes**

- Creació d'una entrada forçada pel més anterior amb 3 tickets resolts

100

- Creació d'una entrada forçada pel més anterior amb 18 tickets resolts

-67

➤ **DW_servei_peticions**

- Creació d'una petició que solliciti el servei SERV005

SERV005

- Cancel·lació d'una petició que solliciti el servei SERV005

SERV003

- Creació de 2 peticions que sollicitin el servei SERV004

SERV004

⇅ SERVEI	⇅ QTT_PETICIONS
SERV003	5
SERV005	5
SERV002	1
SERV004	6
SERV001	5

➤ **DW_incidencies_seu_any**

- Creació de 12 consultes per la seu S003

S003

⇅ ANI	⇅ SEU	⇅ QTT_INCIDENCIES
2018	S003	12
2018	S001	11
2018	S002	1

➤ DW_hores_tickets_L3

Malauradament crear un joc de proves automatitzat és impossible en aquest cas atès que totes les dades de la base de dades es prenen directament del sistema, de manera que l'única manera de testejar aquesta funcionalitat és crear un tiquet assignat a un grup L3, esperar un mínim d'hores i després passar-lo a un estat de Resolt o bé assignar-lo a un grup de Problem Management.

➤ DW_incidencies_PM

- Creció de dos incidències assignades a grups PM

3

- Baixa d'una de les incidències anteriors

2

ANI	QTT_INCIDENCIES
2018	2

➤ DW_sms_no_respostos

- Mirar el percentatge inicial

98

ANI	QTT_TICKETS	QTT_TICKETS_NO_RESPOSTOS
2018	51	50

- Respondre 10 missatges de tickets existents

78

ANI	QTT_TICKETS	QTT_TICKETS_NO_RESPOSTOS
2018	51	40

➤ DW_tickets_tancats

- Canvi d'estat de 3 tiquets existents a Resolt

11

- Cancelacio de 2 tiquets existents

13

QTT_TANCATS
13

➤ DW_tickets_per_canal

- Creació de 5 tiquets per Web a Espanya

```
Telefon : 0
Web : 16,2162162
Xat : 83,7837837
```

- Creació de 2 tiquets per Telefon a Italia

```
Telefon : 18,75
Web : 0
Xat : 81,25
```

- Creació d'un usuari a una seu de França i creació d'un tiquet per Web

```
Telefon : 0
Web : 100
Xat : 0
```

PAIS	MES	ANI	CANAL	QTT_TICKETS_CANAL	QTT_TICKETS_TOTAL
Espanya	5	2018	Web	6	37
Espanya	5	2018	Xat	31	37
Espanya	5	2018	Telefon	0	37
Italia	5	2018	Web	0	16
Italia	5	2018	Xat	13	16
Italia	5	2018	Telefon	3	16
Franca	5	2018	Web	1	1
Franca	5	2018	Xat	0	1
Franca	5	2018	Telefon	0	1