



# Disseny i implementació de la BD d'una plataforma de monitorització de disponibilitat de serveis web

**Joan Garcia Gilabert**

Pla d'estudis: Enginyeria Tècnica d'Informàtica de Gestió

Àrea de treball final: Bases de dades relacionals

**Nom Consultor/a:** Manel Rella Ruiz

**Nom Professor/a responsable de l'assignatura:** Maria Isabel Guitart Hormigo

Data Lliurament

13/6/2016



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

## FITXA DEL TREBALL FINAL

<b>Títol del treball:</b>	Disseny i implementació de la BD d'una plataforma de monitorització de disponibilitat de serveis web
<b>Nom de l'autor:</b>	<i>Joan Garcia Gilabert</i>
<b>Nom del consultor/a:</b>	<i>Manel Rella Ruiz</i>
<b>Nom del PRA:</b>	Maria Isabel Guitart Hormigo
<b>Data de lliurament (mm/aaaa):</b>	<i>Juny / 2016</i>
<b>Titulació o programa:</b>	<i>Enginyeria Tècnica en Informàtica de Gestió</i>
<b>Àrea del Treball Final:</b>	<i>Bases de Dades Relacionals</i>
<b>Idioma del treball:</b>	<i>Català</i>
<b>Paraules clau</b>	<i>monitorització , disponibilitat , serveis web</i>
<p><b>Resum del Treball (màxim 250 paraules):</b> <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i></p>	
<p>La finalitat d'aquest Treball Final de Carrera era el disseny i implementació d'una Base de Dades Relacional per a donar suport a una empresa de Serveis Web. El client volia una aplicació que li permetés gestionar els seus recursos de forma eficient.</p> <p>La Base de Dades resultant d'aquest treball permet: emmagatzemar les dades dels clients, regles de monitorització amb els seus passos, i els registres de monitorització dels Serveis Web (que són el resultat de les execucions periòdiques de les regles de monitorització). El registre de monitorització és un element cabdal d'aquest sistema, doncs recull dades sobre la disponibilitat o no dels serveis. El sistema permet també generar informes i tenir dades mensuals actualitzades on-line en un Datawarehouse.</p> <p>No era objecte d'aquest Treball crear interfícies d'usuari.</p> <p>S'ha seguit la metodologia clàssica de desenvolupament de projectes: desenvolupament en cascada, començant per Anàlisi dels Requeriments. Una etapa prèvia a aquesta que seria el recull dels requeriments, no s'ha fet, doncs ja venien recollits en l'enunciat del TFC.</p> <p>En tractar-se d'un Treball pensat com un cas real, que pogués ser mantingut en el futur, s'ha</p>	

posat molt esforç en documentar molt bé tot el codi ( tant el codi com els seus comentaris s'han fet en anglès ). També s'ha fet especial esforç en el tractament d'errors i excepcions, per tal de tenir una aplicació robusta.

La implementació s'ha fet en Oracle Database 11g ExpressEdition.

Com a conclusió vull destacar que la Base de Dades resultant d'aquest Treball serveix de punt de partida per crear un sistema complet que permeti a l'empresa de Serveis Web la gestió eficient dels recursos. El següent pasos serien:

- a) Dotar la Base de Dades d'una interfície d'usuari
- b) Conectar-la a un motor de monitorització, de manera que els dispositius que fan la monitorització poguessin llegir les regles i passes del nostre sistema, executar-les i crear els registres de monitorització de forma automàtica.

**Abstract (in English, 250 words or less):**

The goal for this Final Thesis was the design and implementation of a Relational Database for a Web Services Enterprise ( Hosting ). The customer wanted an application to efficiently manage resources.

Database implemented in this Final Thesis has the following functionalities: customers information management, handling of Monitoring Rules and Steps and maintenance of a Web Service Monitoring register ( monitoring registers comes from the periodical execution of web service monitoring rules ). Monitoring registers are key elements in this Database because they collect information about Web service availability or not availability. This Database can also generate reports and has a DATAWAREHOUSE with on-line updated information for every month.

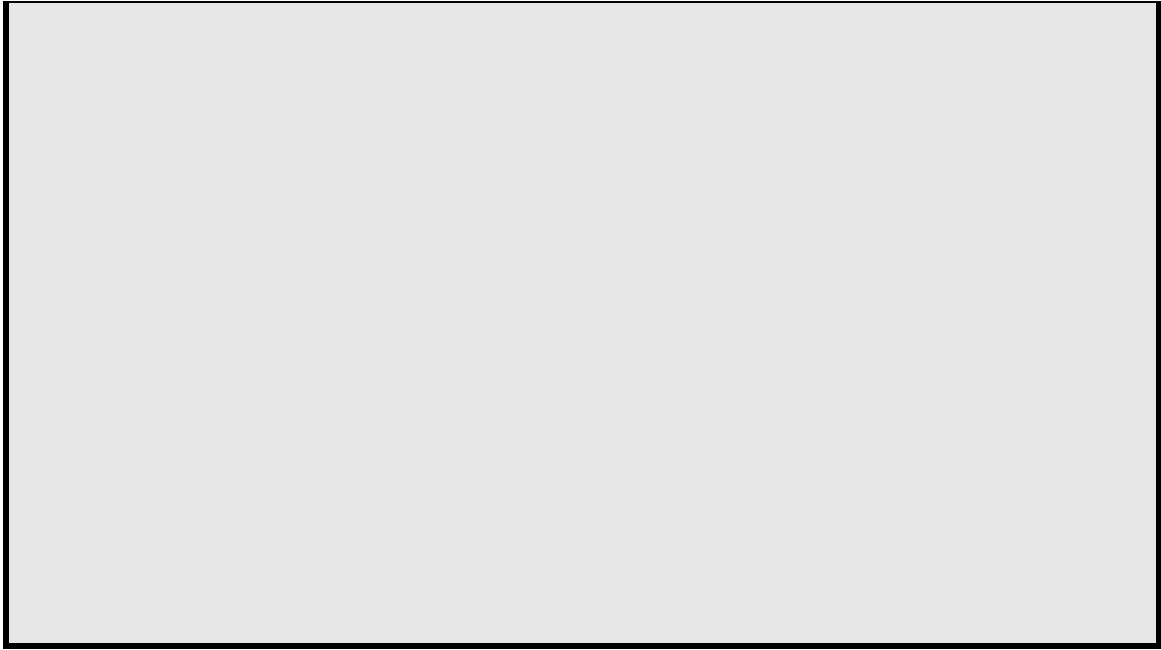
User Interfaces implementation was not a goal for this Final Thesis.

Classical cascade development technique has been used in this Database implementation, starting from the Customer's Requirements Analysis. Previous to this analysis, there should have been the Customer's Requirements Collecting. In this case these requirements were collected in the Final Thesis statement. Due to this final Thesis aims to be a real case, a huge effort has been made to include large documentation through all the code ( code and comments are in English ). An special effort has also been made in errors and exceptions treatment, in order to get a robust application.

Oracle Database 11g Express Edition has been the development application.

As a summary, it has to be point out that the Database delivered with this Final Thesis can be used as the first step to get a complete Web Services management tool. Next steps would be:

- a) Implementing a User interface.
- b) Connecting this Database to a monitoring engine in order to allow monitoring devices to read monitoring rules and steps. These devices would execute these rules and then automatically create monitoring registers.



## ÍNDEX

<b>1</b>	<b>OBJECTE</b> .....	<b>7</b>
<b>2</b>	<b>PLANIFICACIÓ</b> .....	<b>7</b>
<b>3</b>	<b>GESTIÓ DE RISCOS</b> .....	<b>8</b>
<b>4</b>	<b>METODOLOGIA DE TREBALL</b> .....	<b>9</b>
4.1	METODOLIGIA DE TREBALL PREVISTA .....	9
4.2	CANVIS DURANT EL DESENVOLUPAMENT DEL TFC.....	9
<b>5</b>	<b>SOFTWARE</b> .....	<b>10</b>
<b>6</b>	<b>DISSENY CONCEPTUAL</b> .....	<b>10</b>
6.1	RELACIÓ D'ENTITATS .....	10
6.2	DIAGRAMA ENTITAT-RELACIÓ .....	11
6.3	DECISSIONS DE DISSENY .....	11
6.4	RELACIÓ D'ATRIBUTS.....	13
6.5	RESTRICCIONS D'INTEGRITAT .....	19
6.6	PROCEDIMENTS EMMAGATZEMATS ( PACKAGES ) .....	19
<b>7</b>	<b>DISSENY FÍSIC</b> .....	<b>20</b>
7.1	CONVENCIONS DE NOMENCLATURA .....	20
7.2	SEQÜÈNCIES.....	22
7.3	DISPARADORS .....	23
7.4	PROCEDIMENTS EMMAGATZEMATS ( ESPECIFICACIONS ).....	23
7.4.1	<i>Paràmetre de sortida RSP</i> .....	24
7.4.2	<i>PROCEDIMENT LogAnAction_Success</i> .....	24
7.4.3	<i>PROCEDIMENT LogAnAction_Error</i> .....	25
7.4.4	<i>PROCEDIMENT NewWebservice</i> .....	25
7.4.5	<i>PROCEDIMENT DeleteWebService</i> .....	26
7.4.6	<i>PROCEDIMENT UpdateWebService</i> .....	26
7.4.7	<i>PROCEDIMENT NewRule</i> .....	27
7.4.8	<i>PROCEDIMENT DeleteRule</i> .....	28
7.4.9	<i>PROCEDIMENT UpdateRule</i> .....	29
7.4.10	<i>PROCEDIMENT NewStep</i> .....	29

7.4.11	<i>PROCEDIMENT DeleteStep</i> .....	30
7.4.12	<i>PROCEDIMENT UpdateStep</i> .....	31
7.4.13	<i>PROCEDIMENT NewRegister</i> .....	32
7.4.14	<i>PROCEDIMENT DeleteRegister</i> .....	33
7.4.15	<i>PROCEDIMENT UpdateRegister</i> .....	33
7.5	<b>DATAWAREHOUSE</b> .....	34
7.5.1	<i>Funcionament</i> .....	34
7.5.2	<i>El procediment emmagatzemat privat CalculateWarehouseIndicators</i> .....	35
<b>8</b>	<b>TESTING</b> .....	<b>36</b>
<b>9</b>	<b>CONCLUSIONS</b> .....	<b>36</b>
	<b>BIBLIOGRAFIA</b> .....	<b>37</b>
	<b>APÈNDIX A: ERRORS GENÈRICS</b> .....	<b>37</b>
	<b>APÈNDIX B: EXECUCIÓ DE L'SCRIPT DE CREACIÓ DE LA DB</b> .....	<b>37</b>



## **1 OBJECTE**

El propòsit d'aquest Treball Final de Carrera ( TFC ) era el disseny i implementació d'una Base de Dades ( BD ) que resolgués un cas real. No calia incloure l'interfície d'usuari.

La problemàtica resolta se situa en el hosting de serveis web al núvol. Es tractava de fer la implementació d'una BD que permetés a una empresa de hosting monitoritzar la disponibilitat de serveis dels seus clients. Amb aquesta monitorització en un següent desenvolupament, fora de l'abast d'aquest TFC, l'empresa podrà gestionar els seus recursos de la manera més racional possible.

L'empresa de hosting volia que la BD oferís les següents funcionalitats:

1. Emmagatzemar dades per la gestió dels serveis web:
  - a) Client, URL, identificador, descripció, etc...
  - b) Regles de monitorització associades a cada servei web ( amb informació del seus passos corresponents ).
  - c) Alta, baixa i modificació dels serveis web.
2. Obtenció de llistats:
  - a) Donat un any i un mes volien obtenir els serveis web que haguessin tingut registres de monitorització que indiquessin no disponibilitat del servei.
  - b) Per una data conèixer les regles de monitorització actives i els serveis associats.
  - c) Per l'instant actual conèixer els serveis web no disponibles. La regla de monitorització que falla i quan temps porta fallant.
3. Datawarehouse (DW): volien que la BD tingués emmagatzemat en temps real, per cada any i mes :
  - a) El nombre total de registres de monitorització que havien mostrat "disponibilitat de servei".
  - b) El nombre total de registres de monitorització que havien mostrat que "el servei no respon".
  - c) El nombre total de registres de monitorització que havien mostrat que "la resposta del servei web no conté la cadena esperada".
  - d) El nombre de serveis amb disponibilitat entre 95 i 100% i el nombre de serveis entre 0 i 95%.
  - e) El dia del mes amb menys disponibilitat de servei web.
  - f) El client amb menys disponibilitat de servei web durant el mes.
  - g) Identificador de la regla de monitorització que triga més temps en executar-se ( incloent tots els passos ).

## **2 PLANIFICACIÓ**

Per portar a terme el projecte es va fer una planificació seguint les pautes de les PAC.

Cal tenir present que la Redacció Final del TFC es va preveure fer durant el desenvolupament de les PAC.

## Planificació Temporal

Activitat	Inici	Final	Durada dies	
<b>Anàlisi de Requisits</b>	<b>24/02/2016</b>	<b>11/03/2016</b>	<b>16</b>	PAC 1
Lectura Requisits	24/02/2016	27/02/2016	3	
Redacció	27/02/2016	01/03/2016	3	
Instal.lació maquinari	01/03/2016	11/03/2016	10	
<b>Diagrama UML</b>	<b>12/03/2016</b>	<b>01/04/2016</b>	<b>20</b>	PAC 2
Identificació Entitats	12/03/2016	18/03/2016	6	
Identificació de les relacions	18/03/2016	25/03/2016	7	
Diagrama UML	25/03/2016	01/04/2016	7	
<b>Implementació</b>	<b>02/04/2016</b>	<b>20/05/2016</b>	<b>48</b>	PAC 2 PAC 3 F
Implementació de les taules	02/04/2016	09/04/2016	7	
Recull de les restriccions d'integritat	09/04/2016	11/04/2016	2	
Disseny de Procediments emmagatzemats	11/04/2016	25/04/2016	14	
Implementació de Procediments Emmagatzemats	25/04/2016	02/05/2016	7	
Mòdul de Datawarehouse	02/05/2016	20/05/2016	18	
<b>Testing</b>	<b>21/05/2016</b>	<b>03/06/2016</b>	<b>13</b>	F
<b>Redacció Final</b>	<b>04/06/2016</b>	<b>12/06/2016</b>	<b>8</b>	F

### 3 GESTIÓ DE RISCOS

Els dos riscos que es cotemplaven en el desenvolupament d'aquest projecte eren: saturació de feina i problemes de maquinari.

Identificador	Descripció del risc	Probabilitat	Impacte	Mesura de contingència	Classificació del Risc
1	Poca disponibilitat de temps durant la setmana	Alta	Alt	Dedicació total el cap de setaman	Alt
2	Volum de treball més alt de l'esperat	Mitjà	Alt	Dedicar dies de vacances al TFC	Mitjà
3	Problemes de maquinari o programari que comportin pèrdua de	Baixa	Alt	Còpia de seguretat externa	Baix

	feina feta.				
4	Bloqueig de l'ordinador	Baixa	Alt	Duplicar aplicacions a l'ordinador familiar	Baix

Observacions:

- El Risc 1 es va materialitzar, dedicant molts caps de setmana al TFC. Aquest era un risc amb qualificació Alt.
- El Risc 2 que s'havia qualificat de mitjà també es va materialitzar. Es van demanar 3 dies de vacances per tal d'assegurar l'acabament en termini d'un producte robust i buscant la satisfacció personal en tractar-se del TFC.

## 4 METODOLOGIA DE TREBALL

### 4.1 METODOLOGIA DE TREBALL PREVISTA

Es va preveure seguir la metodologia de desenvolupament en cascada.

1. REQUERIMENTS
  - a. Anàlisi de requeriments
2. DISSENY
  - a. Disseny Conceptual de la Base de Dades
3. IMPLEMENTACIÓ
  - a. Implementació física de les taules i les seves relacions
  - b. Inicialització d'algunes de les taules
  - c. Especificació dels procediments emmagatzemats
  - d. Programació dels procediments emmagatzemats
4. TEST
5. DOCUMENTACIÓ
  - a. Afegit de comentaris en tots les mètodes del codi font, tan públics com privats
  - b. Escripció de la memòria

### 4.2 CANVIS DURANT EL DESENVOLUPAMENT DEL TFC

Durant el desenvolupament del TFC s'ha fet canvis respecte a la metodologia prevista:

- A. Hi ha hagut entregues (PACS) intermitges. Amb la qual cosa, la documentació per exemple s'ha anat fent a trossos al llarg de tot el projecte, i no pas tota al final.
- B. La implementació i el test s'han fet en paral·lel. El procés de desenvolupament ha estat el següent:
  - a. Implementació física de les taules (TOTES) i les seves relacions (TOTES)
  - b. Inicialització d'algunes de les taules
  - c. Especificació dels procediments emmagatzemats (TOTS els públics)
  - d. Programació i Test de SERVEIS

- e. Programació i Test de REGLES
- f. Programació i Test de PASSOS
- g. Programació i Test de REGISTRES DE MONITORITZACIO
- h. Programació i Test de DATAWAREHOUSE
- i. Programació i Test de REPORTS

S'hauria d'haver fet un desenvolupament iteratiu amb cicles d'anàlisi, implementació, test i validació amb l'usuari. Però això hauria requerit:

- Més recursos ( temps ).
- Fer més entregues al client ( consultor ).

## 5 SOFTWARE

El programari que s'ha utilitzar en aquest TFC ha estat:

- Diagrames UML: MagicDraw Personal Edition 18.3
- BD:
  - Oracle Database 11g ExpressEdition.
  - Oracle SQL Developer 4.1.3.20

## 6 DISSENY CONCEPTUAL

### 6.1 RELACIÓ D'ENTITATS

Vegis seguidament la relació d'Entitats detectades en el Disseny de la BD més alguna que van aparèixer en fases posteriors de desenvolupament.

**CUSTOMERS:** Entitat auxiliar destinada a incloure les dades de client ( nom, direcció, telèfon, etc... ).

**WEBSERVICETYPES:** Entitat auxiliar que inclou els diferents tipus de serveis web i una descripció genèrica.

**WEBSERVICES:** Entitat que conté les dades dels serveis web. També és l'entitat que guarda l'històric de serveis web, ja que disposa de la possibilitat d'indicar si un servei està o no actiu.

**RULES:** Entitat que conté les regles de monitorització. Inclou tan regles actives com regles que ja no s'executen.

**STEPS:** Aquesta entitat conté informació detallada de les passes de cadascuna de les regles de monitorització.

**REGISTERS:** Entitat que conté les els resultats de les monitoritzacions.

**ERRORS:** Entitat mestre dels possibles errors que podrien produir-se en les execucions dels procediments emmagatzemats.

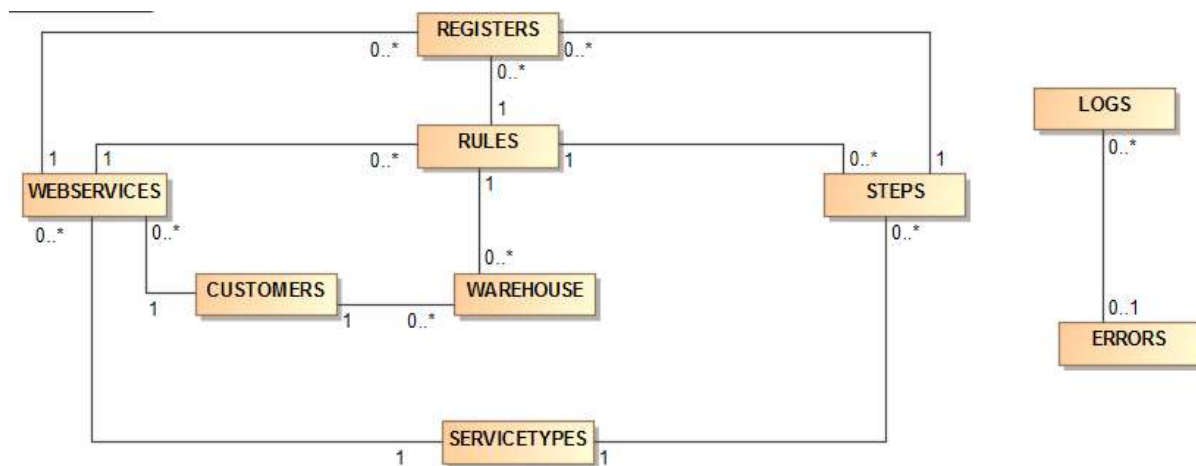
**LOGS:** Entitat típica d'una BD que recull l'activitat de cada usuari. També era un requisit del client en referència als procediments emmagatzemats que calia

enregistrar totes les crides a procediments ( a quin procediment es crida i quins paràmetres d'entrada i sortida s'obtenen ).

**WAREHOUSE:** Aquesta entitat dona resposta al requeriment específic de Datawarehouse (DW). Enmagatzema per a cada mes els indicadors clau que es demanen en els requeriments

## 6.2 DIAGRAMA ENTITAT-RELACIÓ

En base a les entitats detectades i a les funcionalitats requerides en l'enunciat, es va procedir a crear un diagrama UML d'Entitat Relació.



## 6.3 DECISSIONS DE DISSENY

En el Disseny Conceptual es van prendre un sèrie de decisions de disseny que es van anar revisant i refinant durant la implementació. Aquestes decisions de disseny es van anar traslladant al model físic i per tant estan més properes a decisions de disseny físic que conceptual.

1. A totes les entitats i a tots els atributs del model conceptual se'ls va donar noms en anglès. El mateix es va fer amb tots els elements de la implementació física del sistema (taules, camps, restriccions d'integritat, seqüències, triggers, packages, stored procedures, i el codi font que els implementa). Tot es va decidir desenvolupar en anglès
2. Totes les entitat disposen d'una clau primària autogenerada. És un enter sense significat de negoci. Quan s'escau, és a dir, en aquelles entitats que ho permeten, s'ha definit també una clau alternativa, que acostuma a tenir significat de negoci. En el model físic la clau numèrica auto-incremental s'ha implementat amb una seqüència i un trigger associat a cadascuna de les taules. El trigger corresponent a la taula s'executa a cada insert, i el que fa és incrementar la seqüència corresponent i usar el número obtingut com a clau.

3. Aquelles entitats per a les quals s'havia de guardar un històric, s'ha fet, però les dades històriques i actuals s'han guardat en la mateixa taula. Contenen dades tan actuals com històriques ( no vigents) les entitats: WEBSERVICES, RULES, i STEPS. També guarden dades històriques el WAREHOUSE i els REGISTERS, però són casos diferents.
4. Les entitats que tenen atributs booleans del model lògic s'han implementat físicament amb un valor numèric ( 0 per fals i 1 per cert ).
5. L'entitat COSTUMERS és auxiliar i es va implementar de manera senzilla ( amb pocs atributs i sense stored procedures per manipular el contingut ) . A petició del client final, es podria fer tan complicada com fos necessari.
6. A l'entitat WEBSERVICES es guarda l'URL principal del servei web. Les URL secundàries que s'utilitzen en cada cada pas d'una regla de monitorització es guarden en l'entitat STEP.
7. S'ha considerat interessant guardar en cada pas de monitorització el tipus de sol.licitud. Es podria haver optat per pensar que el tipus de sol.licitud va implícit en el tipus de WEBSERVICE al que pertany, però es va pensar que no necessàriament un pas de mntorització havia de pertànyer forçosament a un WEBSERVICETYPE.
8. Tal i com s'ha avançat en la relació d'entitats, l'històric dels serveis es guarda en l'entitat WEBSERVICES junt amb els actuals, distingint-los per mitjà d'un flag d'actiu/no actiu.
9. Tal i com s'ha avançat en la relació d'entitats, l'històric de les regles de monitorització es guarda en l'entitat RULES junt amb les actuals, distingint-los per mitjà d'un flag d'actiu/no actiu.
10. Tal i com s'ha avançat en la relació d'entitats, l'històric dels passos de monitorització es guarda en l'entitat STEPS junt amb els actuals, distingint-los per mitjà d'un flag d'actiu/no actiu.
11. La taula REGISTERS (les execucions de les regles) també guarda dades històriques, però és un cas diferent en el qual no cal tenir el mencionat flag d'actiu/no actiu.
12. Les baixes de serveis, regles i passos es van implementar lògiques i no pas físiques (no hi ha un esborrat físic, sinó que es posa el flag a "inactiu" ). La desactivació és en cascada (desactivant un registre de l'entitat superior desactiva en cadena els registres relacionats de les entitats inferiors). . La raó per a no fer esborrat físic és per que el DW guardava dades històriques que es podien haver de recalculer i referencia a aquests serveis, regles i passos que en el seu dia eren actius.
13. L'esborrat de Registres de Monitorització és físic. Al final però del desenvolupament es va veure que la possibilitat d'esborrar registres de monitorització no tenia interpretació de negoci. Es va implementar un mètode d'esborrat però que només tenia sentit en el cas d'introducció d'un valor erroni.

14. En quant a les longituds dels camps de text, són homogenis en totes les taules:

- a. Noms de persona o d'empresa, adreces de correo electrònic, job titles: límit 50 caràcters.
- b. Adreces postals: límit de 100 caràcters.
- c. Telèfons: límit de 20 caràcters.
- d. URLs, requestes i respostes a les requestes: límit de 1000 caràcters.
- e. Descripcions, paràmetres d'entrada i de sortida de procediments emmagatzemats: límit de 200 caràcters.
- f. Noms de procediments emmagatzemats: límit de 50 caràcters.
- g. En la modificació de Serveis, no es permet que d'un servei se'n canviï el client. Si s'ha de fer això, cal esborrar el servei i crear-ne un de nou.
- h. En la modificació de regles, no es permet que d'una regla se'n canviï el servei. Si s'ha de fer això, cal esborrar el servei i crear-ne un de nou.
- i. En la modificació de passos, no es permet que d'una pas se'n canviï la regla. Si s'ha de fer això, cal esborrar el pas i crear-ne un de nou.

15. Procediments emmagatzemats:

- a. Els noms dels procediments emmagatzemats existents es podrien haver posat en una entitat auxiliar, i guardar-ne algun atribut com ara l'autor del procediment, versió, data de creació, data d'última modificació... No s'ha implementat aquesta taula.
- b. Els procediments emmagatzemats es va preveure presentar-los cadascun en un package segons l'entitat que manipulen. Això va fer que en alguns packages aparaguessin, apart dels procediments d'alta, baixa i modificació, algunes funcions públiques, que calien per a poder consultar dades des d'altres packages. Només en el cas del WAREHOUSE la manipulació es va preveure des d'un altre package ( per deixar clar que la seva manipulació no es feia amb una crida al seu package sino en temps real ). Les dades del WAREHOUSE es calculen des del Package REGISTERS: en afegir, modificar o esborrar, qualsevol REGISTER.

## 6.4 RELACIÓ D'ATRIBUTS

A continuació es fa la relació d'entitats amb els seus atributs.

**CLAU PRIMÀRIA** en negreta.

CLAU ALTERNATIVA subratllada.

*CLAU FORANA* en itàliques.

CUSTOMERS
-----------

En un futur es podria incloure entitats auxiliars (CITY, COUNTRY, ZIPCODES). I també es podria haver afegit també l'atribut CIF, que seria una clau alternativa.

Nom atribut	Tipus	No nul	Comentari
<b>CUSTOMERID</b>	Enter	X	Identificador del client
NAME	Text	X	Nom del client
STREET	Text	X	Adreça postal, nom del carrer
ZIPCODE	Text	X	Adreça postal, codi postal
CITY	Text	X	Adreça postal, nom de la ciutat
COUNTRY	Text	X	Adreça postal, nom del país
EMAIL	Text	X	Adreça de correu
TELEPHONE	Text	X	Número de telèfon
DATECREATION	Data	X	Dia en que es crea el registre

#### WEBSERVICETYPES

Enumeració de tots els tipus de servei web potencialment existents, amb una descripció genèrica per a cadascun d'ells. És una entitat estàtica (un cop inicialitzades les dades, té poc o cap moviment).

Nom atribut	Tipus	No nul	Comentari
<b>TYPEID</b>	Enter	X	Identificador del tipus de servei
DESCRIPTION	Text	X	Descripció genèrica d'aquest tipus de servei
<u>TYPECODE</u>	Text	X	Clau alternativa

#### ERRORS



Entitat auxiliars amb totes els errors possibles. El que es guarda aquí és una descripció genèrica de cada error possible.

L'error ERRORID és un número seqüencial, mentre que en el ERRORCODE (clau alternativa) és el codi d'error d'Oracle (standard).

Per exemple, un registre d'aquesta taula podria ser:

3 ; ORA-00001 ; "Unique constraint violated"

Quan es produeix un error en el registre de LOG es té una descripció detallada de l'error que s'ha produït. Per exemple: 3 ; "Unique constraint WEBSERVICE\_PK violated"

Nom atribut	Tipus	No nul	Comentari
<b>ERRORID_ERR</b>	Enter	X	Enter seqüencial identificador de l'error.
<u>ERRORCODE_ERR</u>	Text	X	
<u>DESCRIPTION_ERR</u>	Text	X	

#### WEBSERVICES

Entitat que guarda els web services i les seves dades bàsiques.

Nom atribut	Tipus	No nul	Comentari
<b>SERVICEID</b>	Enter	X	Identificador del servei
DESCRIPTION	Text	X	Descripció del servei
<i>TYPEID</i>	Enter	X	Tipus del servei.
URL	Text	X	URL principal del servei
<i>CUSTOMERID</i>	Enter	X	Client propietari del servei
ACTIVE	Booleà	X	Flag que indica si el servei està actiu o no. Només es monitoritzen els serveis actius

#### RULES

Regles de monitorització.

Nom atribut	Tipus	No nul	Comentari
<b>RULEID</b>	Enter	X	Identificador de la regla
NAME	Text	X	Nom
<i>SERVICEID</i>	Enter	X	Identificador del servei que aquesta regla monitoritza.
EXECUTEEVERY	Enter	X	Freqüència d'execució de la regla, expressada en segons.

DATECREATION	Data	X	Data que indica quan es va crear la regla.
DATEEND	Data		Data que indica quan la regla s'ha donat de baixa (desactivat). Ja sigui perquè el servei corresponent s'ha esborrat o perquè s'ha creat una regla nova per al servei.
ACTIVE	Booleà	X	Flag que indica si la regla està activa o no. Les regles no actives no s'executen.

### STEPS

Passes de les regles de monitorització (passes de monitorització).

A cada pas es guarda:

- Una URL, que pot o no coincidir amb la URL principal que es guarda en l'entitat WEBSERVICES
- El tipus de sol·licitud (clau forana de WEBSERVICETYPES), que s'ha suposat no té perquè coincidir amb el tipus del servei. (Així per exemple podria tenir un servei de tipus HTTP, que es monitoritzés en varis passos, algun dels quals podria fer requests de tipus HTTP, però potser algun altre pas podria consistir en fer una requesta de tipus SOAP-XML).

Nom atribut	Tipus	No nul	Comentari
<b>STEPID</b>	Enter	X	Identificador del pas.
<u>RULEID</u>	Enter	X	Identificador de la regla de monitorització a la qual aquesta regla pertany.
<u>ORDER</u>	Enter	X	Ordre seqüencial del pas dins la regla de monitorització. Rule Id + Ordre és una clau alternativa de l'entitat, perquè no hi ha mai dos passos de la mateixa regla amb la mateixa numeració.
DESCRIPTION	Text		Descripció del pas. Aquest és dels pocs camps que es permet que pugui ser nul, perquè sovint serà difícil descriure cadascun dels passos.
<i>TYPEID</i>	Enter	X	Tipus de la requesta que s'executa en aquest pas.
URL	Text	X	URL sobre la que s'executa la request.
MAXWAIT	Enter	X	Temps d'espera màxim, en segons.
REQUEST	Text	X	Text de la request a executar.
EXPECTEDANSWER	Text	X	Resposta que s'espera del servidor

### REGISTERS

Registre de les execucions periòdiques de les regles de monitorització ( quan ha tingut lloc l'execució i quin n'ha estat el resultat ).

Nom atribut	Tipus	No nul	Comentari
<b>REGISTERID</b>	Enter	X	Identificador del registre de monitorització
<u>TIMESTAMP</u>	Data	X	Dia i hora (a precisió de segons) de quan s'ha executat el pas. TIMESTAMP i STEPID formen una clau alternativa, en el benentès que no s'executa mai dues vegades el mateix pas en el mateix segon.
YEAR	Enter	X	Any, mes i dia corresponents a l'atribut TIMESTAMP indicat anteriorment. Són per tant atributs redundants, però que els mantenen per facilitar les consultes. (Un requisit del client és que hi ha d'haver un llistat consultant per mes i agrupat per dia. S'ha fet un índex amb aquests tres atributs per a ajudar a fer aquestes consultes de manera eficient). YEAR, MIONTH i DAY es claculen a partir de TIMESTAMP en el procediment emmagatzemat que fa l'alta i modificació de l'entitat REGISTERS.
MONTH	Enter	X	
DAY	Enter	X	
<i>SERVICEID</i>	Enter	X	Identificador del servei monitoritzat
<i>RULEID</i>	Enter	X	Identificador de la regla executada
<u>STEPID</u>	Enter	X	Identificador del pas executat ( amb aquest identificador els altres dos serien redundants, però hagués obligat visitar RULES per a obtenir-los ).
DELAY	Enter	X	Temps de resposta del servei (s'assumeix que en mil·segons).
AVAILABILITY	Enter	X	0 no disponible / 1 valor inesperat / 2 disponible
ERRORID	Text		Cas de que el servei retorni un error, identificador de l'error que ens hagi retornat.
ERRORDescription	Text		Cas de que el servei retorni un error, el text de l'error.

Els camps ERRORID i ERRORDescription, contenen un text o codi d'error del servei del client. Aquest són errors dels serveis, res a veure amb les entitats ERRORS i LOGS que són per errors de l'aplicació d'aquest TFC.

#### LOGS

Recull qui ha executat què, quan i amb quin resultat. Auditoria de les operacions realitzades en el

sistema.			
Nom atribut	Tipus	No nul	Comentari
<b>LOGID</b>	Enter	X	Identificador de l'entrada de log ( nombre seqüencial ).
TIMESTAMP	Data	X	Quan s'ha executat l'acció
USERID	Text	X	Username de l'usuari de la BD que ha executat l'acció
OPERATION	Text	X	Quina acció s'ha executat (nom del procediment emmagatzemat).
PARAMETERSIN	Text		Paràmetres d'entrada del procediment, en l'ordre en que parareixen en l'especificació del procediment emmagatzemat, i separats per ##
PARAMETERSOUT	Text	X	Paràmetres de sortida del procediment, en l'ordre en que parareixen en l'especificació del procediment emmagatzemat, i separats per ##
APPERERRORCODE	Text		Codi de l'error de l'aplicació.
APPERERRORDESCRIPTION	Text		Descripció.
ORAERROR	Text		Codi d'error d'ORACLE.
ORAERRORDESCRIPTION	Text		Descripció.

WAREHOUSE			
Entitat que conté els indicadors mensuals que va demanar el client.			
Nom atribut	Tipus	No nul	Comentari
<b>ID</b>	Enter	X	Identificador ( un número seqüencial ).
<u>YEAR</u>	Enter	X	Any ( >= 2000 )
<u>MONTH</u>	Enter	X	Mes (número de mes, entre 1 i 12) Any i mes formen clau alternativa.
AVAILABLE	Enter	X	Nombre de registres de monitorització del mes que han donat com resultat "servei disponible"
UNAVAILABLE	Enter	X	Nombre de registres de monitorització del mes que han donat com resultat "servei no disponible"

UNEXPECTEDRESPONSE	Enter	X	Nombre de registres de monitorització del mes que han donat de resultat "resposta del servei inesperada"
HIGHAVAILABLE	Enter	X	Nombre de serveis amb disponibilitat alta durant el mes.
LOWAVAILABLE	Enter	X	Nombre del servei amb disponibilitat baixa durant el mes.
BESTDAY	Enter	X	Dia del mes amb millor disponibilitat de serveis.
WORSTDAY	Enter	X	Dia del mes amb pitjor disponibilitat de serveis.
<i>MOSTAFFECTEDCUSTOMERID</i>	Enter	X	El client amb els serveis menys disponibles durant el mes.
<i>SLOWESTRULEID</i>	Enter	X	La regla de monitorització d'execució més lenta durant el mes.

Només hi ha entrades en el WAREHOUSE per un mes si hi ha execucions (REGISTERS) per a aquell mes.

## 6.5 RESTRICCIONS D'INTEGRITAT

Les restriccions d'integritat que es va preveure implementables con constraint varen ser:

- Claus primàries, foranes i valors no nuls.
- Atributs booleans implementats amb els valors 0 i 1 pels atributs ACTIVE ( de WEBSERVICES i RULES ).
- La disponibilitat d'un registre de monitorització a REGISTERS tindrà tres estats ( 0 No disponible, 1 resposta inesperada en la monitorització ó 2 disponible ).
- En l'entitat RULES, l'atribut EXECUTEEVERY ha de ser >0.
- En l'entitat STEPS, l'atribut ORDER ha de ser >0.
- En l'entitat STEPS, l'atribut MAXWAIT ha de ser > 0
- En REGISTER, l'atribut DELAY ha de ser >0
- Els atributs YEAR en diferents entitats, han de ser nombres, posem > 2000
- Els atributs MONTH en diferents entitats han de ser números entre 1 i 12
- Els atributs DAY en diferent entitat, han de ser >1 i no superar el nombre de dies del mes (segons els valors de YEAR i MONTH del mateix registre)

## 6.6 PROCEDIMENTS EMMAGATZEMATS ( PACKAGES )

En l'etapa de disseny es van considerar els següents procediments emmagatzemats ( cadascun implementat en el seu package MONITORINGDB *nom de l'entitat* ). En la apartat 7.4 d'aquesta memòria es pot trobar l'especificació detallada dels procediments que inclou cada package.

- **MONITORINGDB LOGS:** Els procediments emmagatzemats inclosos en aquest package es van preveure per omplenar l'entitat LOGS.
- **MONITORINGDB\_REPORTS:** Els procediments emmagatzemats inclosos en aquest package es van preveure com a resposta a l'exigència del client de tres procediments de consulta:
  - **ReportA:** El client demanava que donat un mes i un any pogués obtenir una sortida de pantalla amb els Serveis Web amb registres de monitorització que indiquen Servei No disponible ( agrupats per dia i ordenats pel temps que el Servei No ha estat disponible ).
  - **ReportB:** El client demanava que donat un dia concret pogués obtenir una sortida per pantalla de les regles de monitorització actives en aquell moment, incloent el servei associat i ordenat per client.
  - **ReportC:** El client demanava poder obtenir una sortida per pantalla amb del serveis no disponibles en aquell instant, indicant la regla i el pas que estan fallant així com el temps que el servei ha estat No disponible.
- **MONITORINGDB SERVICES:** Els procediments emmagatzemats inclosos en aquest package es van preveure per donar d'alta, esborrar o actualitzar serveis web.
- **MONITORINGDB RULES:** Els procediments emmagatzemats inclosos en aquest package es van preveure per crear, esborrar o actualitzar regles de monitorització.
- **MONITORINGDB SERVICES:** Els procediments emmagatzemats inclosos en aquest package es van preveure per donar d'alta, esborrar o actualitzar un servei web.
- **MONITORINGDB STEPS:** Els procediments emmagatzemats inclosos en aquest package es van preveure per donar d'alta, esborrar o modificar el contingut de l'entitat STEPS.
- **MONITORINGDB REGISTERS:** Els procediments emmagatzemats inclosos en aquest package es van preveure per treballar amb l'entitat REGISTERS ( donar d'alta, esborrar o actualitzar un registre ) i actualitzar el contingut de WAREHOUSE.
- **MONITORINGDB TEST:** Els procediments emmagatzemats inclosos en aquest package es van preveure com joc de proves per fer el test de la implementació de BD. Creació, esborrat i modificació de registres de monitorització, regles de monitorització, passos, serveis web, etc...  
**IMPORTANT:** Aquest Package no s'entregaria mai al client. El primer que fa cadascun dels tests és esborrar tota la BD i crear en el seu lloc un joc de proves.

## 7 DISSENY FÍSIC

### 7.1 CONVENCIONS DE NOMENCLATURA

En el disseny físic de la base de dades (DDL) es van seguir unes regles de nomenclatura.

- **Noms de taula:** Els noms de taula, que ja apareixen en el disseny d'entitats, són mots en

anglès explicatius de la informació que conté la taula ( per exemple CUSTOMERS per a la taula de clients ).

Cada taula té un acrònim que es va utilitzar en el nom dels camps. És a dir en la transformació de les entitats en taules es va afegir l'acrònim al final del nom de l'atribut ( COSTUMERID en l'entitat COSTUMERS es va codificar com COSTUMERID\_CST en Oracle ). A continuació la taula d'acrònims utilitzats:

Nom de taula	Acrònim a usar en els noms dels camps
CUSTOMERS	CST
WEBSERVICETYPES	SRT
ERRORS	ERR
WEBSERVICES	SRV
RULES	RUL
STEPS	STP
REGISTERS	REG
USERS	USR
LOGS	LOG
WAREHOUSE	WRH

- **Noms de camps:** Els camps de les taules tenen un nom amb el següent format:

Part1\_Part2Part3

On

Part1: És una paraula explicativa del què és el camp

Part2: Si el camp és clau forana (o forma part d'una clau forana) d'una altra taula, és l'acrònim de tres lletres de la taula referenciada. Si el camp no és clau forana, aquesta part és buida.

Part3: És l'acrònim de tres lletres de la taula a la qual el camp pertany.

Exemples:

TYPEID\_SRT : és l'identificador de tipus de la taula WEBSERVICETYPES.

SERVICEID\_SRVREG: identificador de servei en la taula REGISTERS, que és clau forana de WEBSERVICES.

- **Noms de restriccions de clau primària:** NomdeTaula\_PK.  
Exemple: CUSTOMERS\_PK, restricció que defineix quina és la clau primària de la taula CUSTOMERS.
- **Noms de restriccions de clau alternativa:** NomdeTaula\_AK. Si hi hagués hagut més d'una

clau alternativa s'hagués anomenat: AK1, AK2,...

Exemple: WAREHOUSE\_AK, restricció que defineix quins camps de la taula WAREHOUSE formen clau alternativa.

- **Noms de restriccions de clau forana:** Aquests noms tenen la forma

Part1\_FKPart2

On

Part1: És el nom de la taula

Part2: És l'acrònim de tres lletres de la taula referenciada.

Exemple: WAREHOUSE\_FKRUL, restricció que defineix quines columnes de WAREHOUSE formen clau forana de la taula RULES.

- **Noms de restriccions de check:**

Part1\_CHKPart2

On,

Part1: És un nom de columna.

Part2: És un text indicatiu de la restricció de la que es fa el checking.

NOTNULL

POSITIVE

NOTNEGATIVE

BOOLEAN

Exemple: ACTIVE\_SRV\_CHKBOOLEAN, restricció sobre la columna ACTIVE\_SRV ( de la taula SRV ), que comprova que és la implementació en SQL d'un booleà. Concretament en aquest cas la condició que es comprova és: "ACTIVE\_SRV" IN (0,1).

- **Noms de seqüències:** NomdeTaula\_SEQ
- **Noms de disparadors:** Part1Part2\_TRG

On

Part1 és "NEW"

Part2 el nom de la taula.

Exemple: NEWERROR\_TRG és el disparador que activa la seqüència ERRORS\_SEQ i fa que NEW.ERRORID\_ERR contingui l'últim valor de EORRORID\_ERR+1.

## 7.2 SEQÜÈNCIES

Es va crear una seqüència per taula ja que la clau primària de totes elles és un enter que s'incrementa d'1 en 1.

- COSTUMERS\_SEQ:
- WEBSERVICETYPES\_SEQ
- ERRORS\_SEQ



- WEBSERVICES\_SEQ
- RULES\_SEQ
- STEPS\_SEQ
- REGISTERS\_SEQ
- USERS\_SEQ
- LOGS\_SEQ
- WAREHOUSE\_SEQ

### 7.3 DISPARADORS

Hi ha un disparador per a cada INSERT a les taules de la BD :

- NEWCOSTUMER\_TRG:
- NEWWEBSERVICETYPE\_TRG
- NEWERROR\_TRG
- NEWWEBSERVICE\_TRG
- NEWRULE\_TRG
- NEWSTEP\_TRG
- NEWREGISTER\_TRG
- NEWUSER\_TRG
- NEWLOG\_TRG
- NEWWAREHOUSE\_TRG

### 7.4 PROCEDIMENTS EMMAGATZEMATS ( ESPECIFICACIONS )

Ens referencien aquí només als procediments públics de cada Package i només els principals (Alta, Baixa i Modificació). Alguna funció secundària no queda recollida en aquesta memòria.

**IMPORTANT:** En el text que segueix hi ha taules que descriuen els procediments, però també el codi font conté una descripció molt detallada, en anglès de cadascun del procediments i funcions. En aquest cas, sí es va fer la descripció de tots els procediments, tan públics com privats. Aquesta descripció en el codi inclou: nom del procediment o funció, explicació de la funcionalitat, explicació de cadascun dels paràmetres d'entrada, explicació de cadascun dels paràmetres de sortida, i en el cas de funcions explicació del valor de retorn.

Consideracions:

- Tots els procediments emmagatzemats disposen del paràmetre de sortida RSP que demanava el client.
- Els procediments d'Alta, disposen a més d'un altre paràmetre de sortida que és l'identificador

que el sistema ha donat al nou item.

- Tots els procediments emmagatzemats d'Alta, Baixa, Modificació criden a LogAnAction\_Succes o LogAnAction\_Error. Aquest dos procediments responen al requisit del client que volia un registre de totes les crides a procediments, emmagatzemant paràmetres d'entrada i sortida.

#### **7.4.1 Paràmetre de sortida RSP**

Aquest paràmetre, que era exigència del client, informa del funcionament del procediment emmagatzemat. Conté dos valors possibles:

- “OK” : Si el procediment s’ha executat sense incidents RSP.
- “ERROR\_*descripció de l’error*”: Si s’ha produït un ERROR ens descriu què ha succeït. Els errors són de dos tipus:
  - ERROR exclusiu del procediment ( per exemple: si volem donar d’alta una regla de monitorització que ja està d’alta ens dirà ‘ERROR active rule already exists’ ).
  - ERROR genèric. Els errors genèrics són els que generen un SQLCODE ( introduir una cadena en un atribut numèric, introduir una cadena massa llarga, etc... ). Aquests errors es tracten en un procediment privat PrepareErrorInformation. Aquest procediment genera l’atribut RSP i també els atributs application\_err\_code i application\_err\_msg de la taula de seguiment LOGS ( l’especificació completa d’aquest procediment es troba a l’apèndix B ).

**IMPORTANT:** En cas d’error de l’aplicatiu , on es troba la informació més detallada és en la taula LOGS ( camp APPERRORDESCRIPTION\_LOG ).

#### **7.4.2 PROCEDIMENT LogAnAction Success**

- **Funcionalitat:** Procediment que emmagatzema a LOGS la crida exitosa a un procediment.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_LOGS
- **Paràmetres d’entrada**

userid	NUMBER	Username de l’usuari de BD que realitza l’operació.
action	VARCHAR2	Conté la operació que s’ha portat a terme (és el nom d’algun dels stored procedures d’Alta, Baixa o Modificació).
paramsin	VARCHAR2	Paràmetres en la crida al procediment, en el mateix ordre que en la crida i separats per ##.
paramsout	VARCHAR2	Paràmetres retornats pel procediment, en el mateix ordre que en la crida i separats per ##..

- **Paràmetres de sortida ( no en té )**

### 7.4.3 PROCEDIMENT LogAnAction\_Error

- **Funcionalitat:** Procediment que enmagatzema a LOGS una crida a procediment que ha generat errors.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_LOGS
- **Paràmetres d'entrada**

userid	NUMBER	Identificador d'usuari i PK de USERS.
action	VARCHAR2	Conté la operació que s'ha portat a terme (és el nom d'algun dels stored procedures d'Alta, Baixa o Modificació)
paramsin	VARCHAR2	Paràmetres en la crida al procediment, en el mateix ordre que en la crida i separats per ##.
paramsout	VARCHAR2	Paràmetres retornats pel procediment, en el mateix ordre que en la crida i separats per ##.
errorcode	VARCHAR2	Codi d'error específic del procediment cridat ( error d'aplicatiu ).
errordescription	VARCHAR2	Descripció de l'error específic que s'ha produït ( error d'aplicatiu ).
oracleerror	VARCHAR2	Codi Oracle de l'error d'execució del procediment.
oracleerrordesc	VARCHAR2	Descripció de l'error Oracle que s'ha produït..

- **Paràmetres de sortida ( no en té )**

### 7.4.4 PROCEDIMENT NewWebservice

- **Funcionalitat:** Procediment que permet donar d'alta un Servei Web per a un client específic. Com a paràmetres d'entrada té els atributs de la taula WEBSERVICES ( llevat de la PK SERVICEID ) i retorna el SERVICEID i el paràmetre RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_SERVICES
- **Paràmetres d'entrada**

description	VARCHAR2	Descripció del Servei Web ( no pot ser NULL ).
servicewebsite	NUMBER	Tipus de ServeiWeb (FK de WEBSERVICETYPES ) 1 per HTTP , 2 per SOAP-XML i 3 per REST-JSON

urladdress	VARCHAR2	URL del Servei Web.
costumer	NUMBER	Identificador de client ( FK de CUSTOMERS ).
tmoment	TIMESTAMP	Moment en que es crea el Servei.

Cap d'aquests paràmetres pot ser NULL.

- **Paràmetres de sortida**

serviceid	NUMBER	Identificador del nou Servei Web ( PK de WEBSERVICES ).
RSP	VARCHAR2	els valors d'aquest paràmetre seran:
		OK” : Procediment executat sense incidents.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.5 PROCEDIMENT DeleteWebService

- **Funcionalitat:** Procediment que permet donar de baixa un Servei Web, les seves regles i els seus passos. És un esborrat lògic. Com a paràmetres d'entrada només té SERVICEID ( PK de WEBSERVICES ) i retorna RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_SERVICES
- **Paràmetres d'entrada**

serviceid	NUMBER	Identificador d'un Servei Web ( PK de WEBSERVICES ).
tmoment	<u>TIMESTAMP</u>	Moment en que es dona de baixa el servei.

- **Paràmetres de sortida**

RSP	VARCHAR2	els valors d'aquest paràmetre seran:
		“OK” : Procediment executat sense incidents.
		‘ERROR serviceid was null’ Quan es crida el procediment cal introduir el serviceid.
		‘ERROR service was already removed’ S’ha intentat donar de baixa un servei que havia existit però que ja està de baixa.
		‘ERROR unexpected number of services to delete’ Quan es crida el procediment cal que hi hagi només un Servei Web corresponent al serviceid. Donaria aquest error quan el servei no existeix.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.6 PROCEDIMENT UpdateWebService

- **Funcionalitat:** Procediment que permet actualitzar un Servei Web. Com a paràmetres d'entrada té els atributs de la taula WEBSERVICES ( llevat Customerid ) i retorna el

paràmetre RSP.

- **Tipus:** Públic
- **Package:** MONITORINGDB\_SERVICES
- **Paràmetres d'entrada**

serviceid	NUMBER	Identificador d'un Servei Web ( PK de WEBSERVICES ).
description	VARCHAR2	Descripció del Servei Web ( no pot ser NULL ).
servicewebtype	NUMBER	Tipus de Servei Web ( FK de WEBSERVICETYPES ). 1 per HTTP, 2 per SOAP-XML i 3 per REST-JSON
urladdress	VARCHAR2	URL del Servei Web ( no pot ser NULL ).

- **Paràmetres de sortida**

RSP	VARCHAR2	els valors d'aquest paràmetre seran:
		“OK” : Procediment executat sense incidents.
		‘ERROR serviceid was null’ Quan es crida el procediment cal introduir el serviceid.
		‘ERROR service was deleted’ Quan es crida el procediment per actualitzar un Servei Web, cal que el servei estigui actiu.
		‘ERROR unexpected number of services to update’ Quan es crida el procediment per actualitzar cal que hi hagi només un Servei Web corresponent al serviceid. Donaria aquest error quan el servei no existeix.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.7 PROCEDIMENT NewRule

- **Funcionalitat:** Procediment que permet donar d'alta una Regla de Monitorització per a un Servei Web específic i actiu.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_RULES
- **Paràmetres d'entrada**

rulename	VARCHAR2	Nom de la Regla de Monitorització.
serviceid	NUMBER	Identificador del Webservice al que pertany la Regla ( PK de WEBSERVICES ).
executionfrequency	NUMBER	Freqüència en la que s'ha d'executar la regla ( en segons ). Ha de ser positiu.
tmoment	TIMESTAMP	Moment en que es crea la regla.

Cap dels paràmetres d'entrada por ser NULL.

- **Paràmetres de sortida**

ruleid	NUMBER	Identificador de la nova regla ( PK de RULES )
RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR serviceid was null’: Quan es crida el procediment serviceid no pot ser NULL.
		‘ERROR service do not exist’: Quan es crida el procediment el Serveiweb ha d'estar a la taula WEBSERVICES.
		‘ERROR service was deleted’: Quan es crida el procediment el Serveiweb ha d'estar actiu.
		‘ERROR active rule already exists’: Quan s'ha cridat el procediment per donar d'alta una regla, aquesta regla no pot estar ja d'alta.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.8 PROCEDIMENT DeleteRule

- **Funcionalitat:** Procediment que permet fer l'esborrat d'una Regla de Monitorització. És un esborrat lògic ( es desactiva la regla ). Només retorna RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_RULES
- **Paràmetres d'entrada**

ruleid	NUMBER	identificador d'una regla de monitorització ( PK de RULES )
Tmoment	TIMESTAMP	indica el moment en que s'ha desactivat la regla.

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR ruleid was null’: Quan es crida el procediment ruleid no pot ser NULL.
		‘ERROR rule was already removed’: Quan es crida el procediment la regla ha d'estar activa.
		‘ERROR unexpected number of rules to delete ’: Quan es crida el procediment i no existeix un regla amb el ruleid introduït o si hi hagués més d'una regla amb el mateix ruleid.
		ERRORS genèrics ( veure APÈNDIX A ).

### 7.4.9 PROCEDIMENT UpdateRule

- **Funcionalitat:** Procediment que permet actualitzar una regla de monitorització. Només retorna RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_RULES
- **Paràmetres d'entrada**

ruleid	NUMBER	Identificador d'una regla de monitorització ( PK de RULES ).Aquest paràmetre no pot ser nul.
name	VARCHAR2	Nom que volem donar a la regla ( no pot ser NULL ).
executionfrequency	NUMBER	Freqüència en la que s'ha d'executar la regla ( en segons ). Aquest paràmetre ha de ser >0 i no pot ser NULL.

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR ruleid was null’: Quan es crida el procediment ruleid no pot ser NULL.
		‘ERROR rule was deleted’: Quan es crida el procediment la regla ha d'estar activa.
		‘ERROR unexpected number of rules to delete ’: Quan es crida el procediment i no existeix un regla amb el ruleid introduït o hi ha més d'una regla amb el mateix ruleid.
		ERRORS genèrics ( veure APÈNDIX A ).

### 7.4.10 PROCEDIMENT NewStep

- **Funcionalitat:** Procediment que crea un nou pas per una regla de monitorització.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_STEPS
- **Paràmetres d'entrada**

ruleid	NUMBER	Identificador d'una regla de monitorització ( PK de RULES ) . No pot ser NULL.
steporder	NUMBER	Indica l'ordre seqüencial del pas dins la regla de monitorització. No pot ser NULL ni repetir-se.
description	VARCHAR2	Descripció textual de que la regla de monitorització en aquest pas. Pot ser NULL ( en previsió de regles molt

		complexes ).
servicetype	NUMBER	Identificador del tipus de Servei Web. 1 per HTTP, 2 per SOAP-XML i 3 per REST-JSON.
urladdress	VARCHAR2	URL sobre la que es fa la consulta ( no pot ser NULL ). No pot ser NULL. El sistema no verifica que la URL sigui vàlida.
maxwait	NUMBER	Temps màxim d'espera en la resposta. No pot ser NULL i ha de ser positiu.
request	VARCHAR2	Text de la consulta a realitzar ( no pot ser NULL ).
expectedanswer	VARCHAR2	Resposta esperada del servidor identificat per la URL ( no pot ser NULL ).
tmoment	TIMESTAMP	Moment de creació del pas.

- **Paràmetres de sortida**

stepid	NUMBER	Codi identificar del pas ( PK d'STEPS ).
RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR ruleid was null’: Quan es crida el procediment ruleid no pot ser NULL.
		‘ERROR rule does not exist’: Quan es crida el procediment la regla ha d'existir.
		‘ERROR rule was deleted’: Quan es crida el procediment la regla havia estat esborrada.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.11 **PROCEDIMENT DeleteStep**

- **Funcionalitat:** Procediment que permet esborrar el pas d'una regla de monitorització ( esborrat lògic i es fa desactivant la regla ). Només retorna RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_STEPS
- **Paràmetres d'entrada**

stepid	NUMBER	Identificador del pas ( PK d'STEPS ). No pot ser NULL.
tmoment	TIMESTAMP	Moment en que es fa l'esborrat.

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
-----	----------	--



		“OK” : Procediment executat sense incidents.
		‘ERROR stepid was null’: Quan es crida el procediment stepid no pot ser NULL.
		‘ERROR step was already removed’: Quan es crida el procediment el pas ha d’estar actiu.
		‘ERROR unexpected number of steps to delete’: Quan es crida només pot haver un pas amb identificador stepid.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.12 **PROCEDIMENT UpdateStep**

- **Funcionalitat:** Procediment que permet modificar un pas d’una regla de monitorització. Només retorna el paràmetre RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_STEPS
- **Paràmetres d’entrada**

stepid	NUMBER	Identificador del pas d’una regla de monitorització ( PK d’STEPS ). No pot ser NULL.
steporder	NUMBER	Indica l’ordre seqüencial del pas dins la regla de monitorització. Una regla de monitorització no pot tenir dos passos amb el mateix ordre. No pot ser NULL.
description	VARCHAR2	Descripció textual de que la regla de monitorització en aquest pas. Pot ser NULL.
servicetype	NUMBER	Indica quin tipus de consulta es farà en aquest pas. 1 per HTP, 2 per SOAP-XML i 3 per REST-JSON.
urladdress	VARCHAR2	URL sobre la que es fa la consulta ( no pot ser NULL ). El sistema no verifica que sigui una URL vàlida.
maxwait	NUMBER	Temps màxim d’espera en la resposta ( en segons ). Ha de ser >0 i no pot ser NULL.
request	VARCHAR2	Text de la consulta a realitzar la servidor URL.
expectedanswer	VARCHAR2	Resposta esperada del servidor.

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d’aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR stepid was null’: Quan es crida el procediment stepid no pot ser NULL.

		'ERROR step was deleted': Quan es crida el procediment el pas ha d'estar actiu.
		'ERROR unexpected number of steps to update': Quan es crida només pot haver un pas amb identificador stepid.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.13 **PROCEDIMENT NewRegister**

- **Funcionalitat:** Procediment que afegix un Registre de Monitorització a la taula REGISTERS. Els paràmetres d'entrada són tots els atributs d'un Registre de Monitorització ( llevat de REGITERID el PK de REGISTERS, Ruleid, Serviceid i Year, Month i Day que es calculen ). Els paràmetres de sortida són REGITERID i RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_REGISTERS
- **Paràmetres d'entrada**

stepid	NUMBER	Clau alternativa i forana de REGISTERS que identifica el pas que s'ha executat.
answerdelay	NUMBER	Temps que ha trigat en executar-se la regla de monitorització fins el pas en qüestió. És >0 i no pot ser NULL.
available	NUMBER	Indica si el Servei Web estava disponible. 0 - No disponible 1 –Resposta inesperada 2 – Disponible.
errorid	VARCHAR2	És el codi d'error que retorna el Servei Web.
errordescription	VARCHAR2	Descripció de l'error retornat pel Servei Web.
tmoment	TIMESTAMP	Indica el moment en que s'ha executat el pas que es registra.

- **Paràmetres de sortida**

registerid	NUMBER	Identificador únic assignat pel sistema ( PK de la taula REGISTERS ).
RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		"OK" : Procediment executat sense incidents.
		'ERROR stepid was null': Quan es crida el procediment stepid no pot ser NULL.
		'ERROR step does not exist': Quan es crida el procediment per Registra un pas, aquest ha d'existir.

		'ERROR step was deleted': Quan es crida el procediment el pas ha d'estar encara actiu.
		'ERROR moment was null'. Quan es crida el procediment el temps en que s'ha executat la monitorització no pot ser null.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.14 **PROCEDIMENT DeleteRegister**

- **Funcionalitat:** Procediment que esborra un Registre de Monitorització de la taula REGISTERS. Només cal indicar REGISTERID ( PK de REGISTERS ). El paràmetre de sortida és REGISTERID.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_REGISTERS
- **Paràmetres d'entrada.**

registerid	NUMBER	Identificador d'un Registre de monitorització ( PK de REGISTERS ).
------------	--------	--

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		"OK" : Procediment executat sense incidents.
		'ERROR registerid was null': Quan es crida el procediment registerid no pot ser NULL.
		'ERROR unsuspected number of monitoring registers to delete': Quan es crida el procediment només pot haver un registre amb identificador registerid.
		ERRORS genèrics ( veure APÈNDIX A ).

#### 7.4.15 **PROCEDIMENT UpdateRegister**

- **Funcionalitat:** Procediment que permet modificar un Registre de Monitorització a la taula REGISTERS. Els paràmetres d'entrada són tots els atributs d'un Registre de Monitorització ( llevat de Serviceid i Year, Month and Day que es calculen ). Els paràmetre de sortida és RSP.
- **Tipus:** Públic
- **Package:** MONITORINGDB\_REGISTERS
- **Paràmetres d'entrada**

registerid	NUMBER	Identificador del Registre de Monitorització ( PK de REGISTERS ).
tmoment	TIMESTAMP	Clau alternativa de REGISTERS i que indica el moment en

		que s'ha executat el pas que es registra.
stepid	NUMBER	Clau alternativa i forana de REGISTERS que identifica el pas que s'ha executat.
answerdelay	NUMBER	Indica el temps que ha trigat el servidor en contestar.
available	NUMBER	Indica si el Servei Web estava disponible. 0 - No disponible 1 -Valor inesperat 2 – Disponible.
errorid	VARCHAR2	És el codi d'error que retorna el Servei Web.
Errordescription	VARCHAR2	Descripció de l'error retornat pel Servei Web.

- **Paràmetres de sortida**

RSP	VARCHAR2	els diferents valors d'aquest paràmetre són:
		“OK” : Procediment executat sense incidents.
		‘ERROR registerid was null’: Quan es crida el procediment registerid no pot ser NULL.
		‘ERROR stepid was null’: Quan es crida el procediment stepid no pot ser NULL.
		‘ERROR step does not exist’: Quan es crida el procediment per modificar un Registre el pas ha d'existir.
		‘ERROR step was deleted’: Quan es crida el procediment el pas ha d'estar encara actiu.
		‘ERROR moment was null’. Quan es crida el procediment el temps en que s'ha executat la monitorització no pot ser null.
		‘ERROR unexpected number of monitoring registers to update’. Quan es vol actualitzar un Registre de Monitorització cal que n'hi hagi un i només un.
		ERRORS genèrics ( veure APÈNDIX A ).

## 7.5 DATAWAREHOUSE

Per donar resposta a la demanda del client d'un dispositiu de DATAWAREHOUSE es va preveure la taula WAREHOUSE, alimentada pels procediments emmagatzemats inclosos en el MONITORINGDB\_REGISTERS.

### 7.5.1 Funcionament

El package MONITORINGDB\_REGISTERS disposa del procediments emmagatzemats públics:

- NewRegister
- DeleteRegister
- UpdateRegister

Després d'afegir, esborrar o modificar un registre de la taula REGISTERS criden al procediment emmagatzemat privat CalculateWarehouseIndicators. Aquest procediment actualitza la taula WAREHOUSE. D'aquesta manera s'assegura que en tot moment la taula disposa de la informació actualitzada. Amb un SELECT sobre WAREHOUSE es podrà respondre en qualsevol moment al requeriment del client d'informació mensual: registres de serveis disponibles, no disponibles, grau de disponibilitat, dia del mes amb menys disponibilitat, client amb menys servei i regla de monitorització més lenta.

### 7.5.2 El procediment emmagatzemat privat CalculateWarehouseIndicators.

- **Funcionalitat:** Aquest procediment emmagatzemat privat és cridat pels procediments NewRegister, DeleteRegister i UpdateRegister un cop aquests procediments han finalitzat i si ho han fet amb èxit. NewRegister i DeleteRegister fan una crida a CalculateWarehouseIndicators i UpdateRegister en pot fer una o dues ( en farà dues si s'ha modificat la data d'un registre i resultan en l'eliminació d'un registre d'un mes i l'afegit al d'un altre ).
- **Tipus:** Privat
- **Package:** MONITORINGDB\_REGISTERS
- **Paràmetres d'entrada:**

tyear	NUMBER	any del registre de monitorització afegit, esborrat o actualitzat.
tmonth	NUMBER	mes del registre de monitorització afegit, esborrat o actualitzat.

- **Paràmetres de sortida ( no en té )**

- **Pseudocodi:**

Paràmetres d'entrada: any, mes del registre de monitorització nou, esborrat o afegit.

Si

s'ha esborrat de REGISTERS l'únic registre que hi ha a la taula WAREHOUSE del mes

aleshores

esborrar registre ( mes , any ) de WAREHOUSE

sino

Recalculer els indicadors del WAREHOUSE pel mes ( disponible, no disponible, resposta inesperada, més disponibilitat, menys disponibilitat, regla més lenta,...)

Si no existeix al WAREHOUSE un registre ( mes, any )

aleshores

inserir registre ( mes, any ) a WAREHOUSE

sino

actualitzar registre ( mes , any ) amb els indicadors recalculats

fisifisi**8 TESTING**

Es va posar especial ènfasi en crear procediments enmagatzemats per fer el test de l'aplicació. Aquests procediments es van agrupar en el package MONITORINGDB\_TEST.

Com s'ha comentat anteriorment aquest package no s'entregaria al client perquè esborra la Base de Dades per poder procedir als tests ( Initialize\_Database\_Content ).

Al igual que en tota l'aplicació s'ha documentat el codi per si els Test poguessin ser útils en un futur desenvolupament.

Els procediments enmagatzemats inclosos en MONITORINGDB\_TEST són:

- Run\_TESTCASE\_01\_NEWWEBSERVICE
- Run\_TESTCASE\_02\_UPDWEBSERVICE
- Run\_TESTCASE\_03\_DELWEBSERVICE
- Run\_TESTCASE\_04\_NEWRULE
- Run\_TESTCASE\_05\_UPDATERULE
- Run\_TESTCASE\_06\_DELETERULE
- Run\_TESTCASE\_07\_NEWSTEP
- Run\_TESTCASE\_08\_UPDATESTEP
- Run\_TESTCASE\_09\_DELETESTEP
- Run\_TESTCASE\_10\_NEWREGISTER
- Run\_TESTCASE\_11\_UPDATEREGISTER
- Run\_TESTCASE\_12\_DELETEREGISTER
- Run\_TESTCASE\_13\_REPORTA
- Run\_TESTCASE\_14\_REPORTB
- Run\_TESTCASE\_15\_REPORTC
- Run\_TESTCASE\_16\_WAREHOUSE
- Run\_ALL\_TESTCASES

**9 CONCLUSIONS**

- S'ha fet la implementació sol.licitada pel client.
- S'ha donat molta importància a la documentació del codi. Seguint les indicacions del client s'ha documentat el codi per tal que pugui ésser reutilitzat.
- S'ha dedicat molt temps al tractament d'errors i correcte omplenat de la taula LOGS.
- S'ha implementat el Datawarehouse. Es va valorar altres implementacions però per simplicitat

es va seguir la recomanació d'alimentar el Datawarehouse amb els procediments emmagatzemats prèviament codificats.

- S'ha dedicat molt esforç en el Package MONITORINGDB\_TEST per garantir que el codi funciona correctament.
- Aquesta BD pot servir per alimentar sistemes que gestionin els Servidors de l'empresa de Hosting.

### **BIBLIOGRAFIA**

- Manuals d'Oracle en format pdf:
  - "e18147 2 Day Developer's Guide.pdf"
  - "b28286 Oracle DATABASE SQL Language Reference.pdf"
- Materials didàctics de les assignatures: de Bases de dades I i Bases de dades II.
- **Perdita Stevens with Rob Pooley** "Using UML Software engineering with objects and components", Addison-Wesley, 2000. ( ISBN 0-201-64860-1 )
- **Alberto Abelló Gamazo** "Disseny multidimensional", ©FUOC PID\_00189746

### **APÈNDIX A: ERRORS GENÈRICS**

Els errors que figuren en aquest Apèndix A els pot contenir el paràmetre RSP de sortida dels procediments emmagatzemats. Aquests errors són propis d'Oracle i en ser comuns a tots els procediments emmagatzemats es recullen en un sòl paràgraf d'aquesta memòria. Els errors propis de cada procediment es descriuen en l'apartat 7.5 en l'especificació de cada procediment públic.

<b>SQL CODE</b>	<b>RSP</b>	<b>Observacions</b>
1400	ERROR cannot insert null	
1407	ERROR cannot update to null	
2290	ERROR check constraint violated	
2291	ERROR parent key not found	
12899	ERROR value too large for column	
	ERROR unknown error	En cas que l'SQL CODE no sigui cap dels anteriors

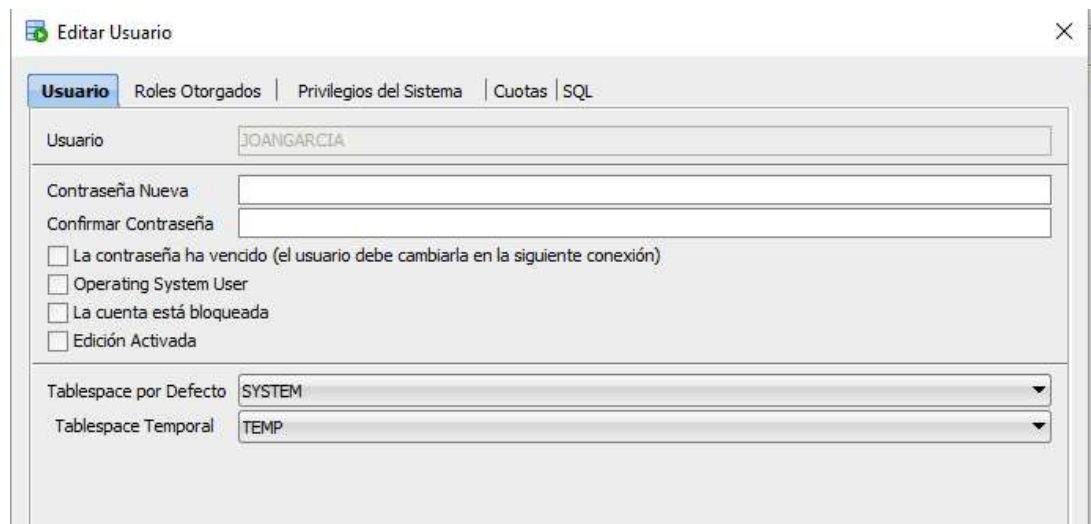
### **APÈNDIX B: EXECUCIÓ DE L'SCRIPT DE CREACIÓ DE LA DB**

L'script de creació de la Base de Dades és al fitxer:

export\_20160611\_1952.sql

Els passos per fer la instal.lació són els següents:

1. Iniciar l'Oracle SQL Developer.
2. Amb permisos d'administrador, crear l'usuari JOANGARCIA.
  - a. Valors a la pestanya "Usuario"

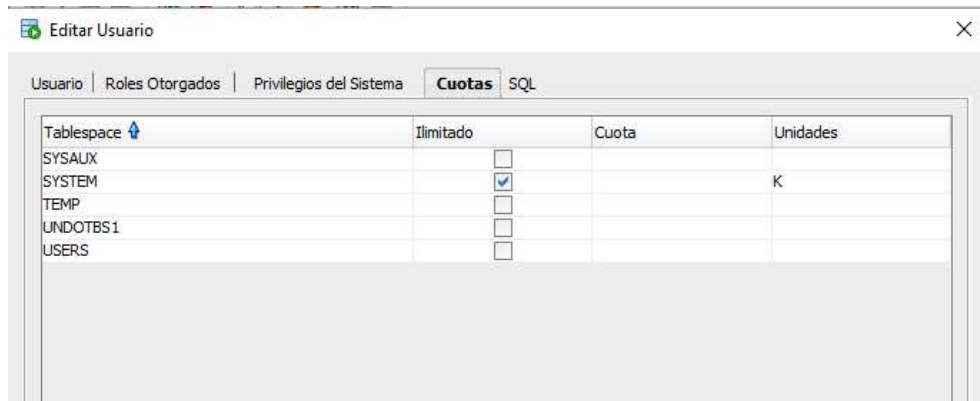


- b. Roles Otorgados ( ignorar )
- c. Privilegios del Sistema: calen com a mínim els següents:
  - i. ALTER\_ANY\_SEQUENCE
  - ii. COMMENT\_ANY\_TABLE
  - iii. CREATE\_ANY\_SEQUENCE
  - iv. CREATE\_ANY\_TABLE
  - v. CREATE\_ANY\_TRIGGER
  - vi. CREATE\_PROCEDURE
  - vii. CREATE\_SEQUENCE
  - viii. CREATE\_SESSION
  - ix. CREATE\_TABLE
  - x. CREATE\_TABLESPACE
  - xi. CREATE\_TRIGGER
  - xii. DELETE\_ANY\_TABLE
  - xiii. EXECUTE\_ANY\_PROCEDURE
  - xiv. IMPORT\_FULL\_DATABASE
  - xv. INSERT\_ANY\_TABLE
  - xvi. SELECT\_ANY\_SEQUENCE
  - xvii. SELECT\_ANY\_TABLE



xviii. UPDATE\_ANY\_TABLE

d. A la pestanya “Cuotas”:



e. Aplicar y Cerrar

3. Desconnectar de la BD.
4. Obrir l'script “export\_20160611\_1952.sql” amb el Notepad.
5. Copiar tot el contingut al portapapers.
6. Connectar-se a la BD amb l'usuari JOANGARCIA.
7. Enganxar el contingut del portapapers a la finestra d'SQL.
8. Executar l'script.