

# **TFC BD Relacionals**

## **Sistema Gestió d'amonestacions i sancions en centres educatius**

**Gabriel Duran Vaquer**  
ETIG

**Consultor : Jordi Ferrer Duran**

**12 de juny del 2011**

A na Victoria, pels ànims incondicionals,  
i al meus fills Biel i Maria,  
perquè les sigui un exemple de constància.

## TREBALL FINAL DE CARRERA

### BASES DE DADES RELACIONALS

La Generalitat de Catalunya, com a projecte de millores dins l'ensenyança, ha decidit preparar una implementació d'un sistema de gestió d'amonestacions i sancions en els centres educatius. Aquest sistema, serà un complement d'altres gestions ja existents als centres educatius, els quals tenen altres funcions diferents a la que es pretén al nou projecte.

Per realitzar aquest projecte, la Generalitat de Catalunya ha obert, en una primera fase, un concurs públic per rebre propostes per damunt el disseny de una base de dades que serà utilitzat al nou sistema. En una propera fase, es té previst obrir un altre concurs públic, per la realització de l'aplicació de gestió, que utilitzarà la base de dades per realitzar el sistema de gestió d'amonestacions i sancions.

La Generalitat de Catalunya presenta un plec tècnic de condicions, que han de complir totes les propostes per ser avaluades.

Aquest treball final de carrera és una proposta per donar una solució vàlida a les necessitats esmentades i cobrint totes les especificacions tècniques, incloses al plec tècnic de condicions.

Aquest document, com a memòria del treball final de carrera, analitzarà les necessitats del projecte i marcarà els objectius a dur a terme. Aquests objectius engloben, la realització del disseny lògic de la base de dades mitjançant un diagrama entitat-relació i la seva descripció conceptual. També es detalla els procediments necessaris per la correcta gestió de la base de dades segons les condicions tècniques.

Adjunt aquest document, hi ha els scripts de creació del contingut de la base de dades: taules, index, disparadors i procediment. Juntament amb un document explicatiu del orde de creació.

També s'adjunta un joc de proves, per comprovar el correcte funcionament del producte.

El Sistema de Gestió de Base de Dades utilitzat ha estat Oracle Database Express Edition 10g, per tant, la solució aportada està en llenguatge PL/SQL.

## INDEX

1 TFC BD Relacionals	7
1.1 Introducció	7
1.2 Justificació del TFC	7
1.3 Objectius del TFC	7
1.4 Metodologia	8
1.5 Eines a utilitzar	9
1.6 Planificació del projecte	9
1.7 Productes Obtinguts	11
2 Desenvolupament BD	12
2.1 Introducció	12
2.2 Disseny conceptual BD	13
2.2.1 Esquema Entitat-Relació	13
2.2.2 Atributs de les entitats	14
2.2.3 Justificació de la proposició realitzada	15
2.2.3.1 Generalitzacions / Especialitzacions	15
2.2.3.2 Entitats associatives	16
2.2.3.3 Interrelacions, connectivitat i dependència	16
2.3 Disseny lògic BD	20
2.3.1 Entitats	20
2.3.2 Interrelacions	22
2.4 Scripts BD	23
2.4.1 Ajusts del disseny	23
3 Implementació BD	25
3.1 Introducció	25
3.2 Taules de persistència per l'estadística	25
3.3 Logs	26
3.4 Procediments	26

3.4.1 Introducció als procediments	26
3.4.2 Esquema de procediments	27
3.4.2.1 Manteniment de taules de persistència	27
3.4.2.1.1 Procediment per la inserció de dades	27
3.4.2.1.2 Procediment per la modificació de dades	29
3.4.2.1.3 Procediment per la eliminació de dades	31
3.4.2.2 Control de sancions	34
3.4.2.2.1 Procediment assignació automàtica de sancions	34
3.4.2.2.2 Procediment actualitzar amonestacio-sancio.	35
3.4.2.3 Llistats dades bàsiques	36
3.4.2.3.2 Llistat alumnes d'un curs	36
3.4.2.3.3 Llistat tipus d'amonestacions	38
3.4.2.3.4 Llistat tipus de sancions	38
3.4.2.3.5 Llistat de amonestacions d'un alumne	39
3.4.2.3.6 Llistat de totes les amonestacions	40
3.4.2.3.7 Llistat de sancions d'un alumne	41
3.4.2.4 Mòdul estadístic	42
3.4.2.4.1 Procediment p_estadistica_e1	42
3.4.2.4.2 Procediment p_estadistica_e2	43
3.4.2.4.3 Procediment p_estadistica_e3	44
3.4.2.4.4 Procediment p_estadistica_e4	44
3.4.2.4.5 Procediment p_estadistica_e5	45
3.4.2.4.6 Procediment p_estadistica_e2_e5	46
3.4.2.4.7 Procediment p_estadistica_e6	47
3.4.2.4.8 Procediment p_estadistica_e6_add	48
3.4.2.4.9 Procediment p_estadistica_e7	49
3.4.2.4.10 Procediment p_estadistica_e8	50
3.4.3 Errors reportats pels procediments.	51
3.4.4 Implementació de procediments	51

3.4.4.1 Manteniment de taules de persistència	51
3.4.4.2 Control de sancions	53
3.4.4.3 Llistat dades bàsiques	54
3.4.4.4 Mòdul estadístic	54
4 Producte final i jocs de proves	55
5 Cost del projecte	56
6 Conclusions	56
7 Glossari	56
8 Bibliografia	57
9 Annexos	57

## **1. TFC BD Relacionals**

### **1.1 Introducció**

El treball final de carrera és la culminació de qualsevol carrera, en aquest cas de l'Enginyeria Tècnica d'Informàtica Gestió.

L'Enginyeria Tècnica d'Informàtica Gestió és una titulació oficial que té com objectiu la formació de professionals en el camp de l'anàlisi de projectes informàtics, la programació, la gestió i el manteniment de sistemes d'informació i com no l'estructuració, gestió i manteniment de dispositius d'emmagatzematge físics i lògics.

L'ETIG és una carrera en extinció debut a la implantació del anomenat pla de Bolonya, el qual té el fi de unificar els criteris d'avaluació dins l'ensenyament del espai europeu. El Grau d'Informàtica és la titulació que substitueix a l'ETIG unificant-lo amb l'Enginyeria Tècnica de Sistemes i l'Enginyeria Superior.

### **1.2 Justificació del TFC**

El treball final de carrera de base de dades relacionals té com objectiu el desenvolupament, d'una forma pràctica, del que s'ha après durant la carrera en el món de les bases de dades.

Les assignatures cursades Base de Dades I i II, Minería de Dades i Estructura de la Informació han fomentat, amb prou solidesa, els ciments per realitzar un treball d'aquesta mena.

Les bases de dades són molt importants dins el camp professional de la informàtica, totes les ERP és basen en Sistemes de Gestió de emmagatzematge, qualsevol sistema de consulta de informació o gestió del coneixement tenen a les bases de dades com a peça fonamental, les pàgines web dinàmiques s'alimenten d'informació emmagatzemada; aquests són exemples genèrics, i no únics, on especialistes en el camp de les bases de dades són necessaris, i és un motiu, més que suficient, per realitzar un treball de final de carrera dins aquesta àrea.

### **1.3 Objectius del TFC**

La proposta del treball de final de carrera, dins aquest semestre, és la d'implementar una solució per cobrir les necessitats, sorgides a la Conselleria d'Ensenyament del nou govern de la Generalitat de Catalunya, que han sorgit al voler desenvolupar una aplicació per gestionar les amonestacions i sancions al centres educatius d'ensenyament secundari.

El projecte encarregat consisteix en dissenyar i crear una base de dades, que utilitzarà la futura aplicació com emmagatzematge de la informació.

L'anàlisi funcional de la base de dades, entregat amb l'enunciat del treball final de carrera, detalla els requisits funcionals que ha de tenir l'aplicació, segons el plec tècnic del concurs que ha tret la Generalitat de Catalunya. Aquest anàlisi engloba els següents punts:

- El sistema de gestió ha de emmagatzemar les dades bàsiques dels alumnes i professors, així com els cursos i assignatures en el qual estan matriculats i imparteixen respectivament.
- Ha de guardar les hores de tutoria dels professors que tenen d'atenció als pares i les de consultoria als alumnes.
- Manteniment del calendari escolar, incloent els dies no lectius.
- Emmagatzematge dels tipus d'amonestacions i dels tipus de sancions. Inicialment hi haurà una càrrega de tipus ja predefinits.
- Emmagatzematge de les amonestacions que els professors fan als alumnes.
- Emmagatzematge de sancions i sistema d'assignació automatitzada de sancions per tipus d'amonestació.
- La base de dades disposarà de procediments per realitzar les altes, baixes i modificacions de totes les taules definides.
- La base de dades tindrà definits els procediments necessaris per realitzar les consultes relatives a alumnes, tipus d'amonestacions i sancions, a més de les amonestacions i sancions imposades.

Aquests darrers procediments englobaran els procediments per el mòdul estadístic, els quals tindran una configuració per oferir les dades en temps constant 1.

El mòdul estadístic ha de respondre, com a mínim, als següents requisits:

- Número amonestacions per alumne
- Número sancions per alumne i any
- Mitjana d'amonestacions per professor i any
- Número de sancions per curs i any
- Alumne amb més sancions per un any concret
- Professor que més amonestacions ha posat per un curs concret
- Mitjana de sancions que tenen els alumnes per curs
- Número d'alumnes sense amonestacions

#### **1.4 Metodologia**

Es dissenyarà la base de dades d'acord amb les especificacions detallades al anàlisi funcional del punt 1.3.

Es realitzarà un diagrama E/R per obtenir el model conceptual del disseny de la base de dades.

Es crearà els *scripts* de sql de creació de taules, índex i disparadors.

El disseny i creació de la base dades ha de complir la forma normal *Boyce-Codd* (FNBC).

Implementació dels procediments necessaris segons els requisits exigits, tenen en compte el tractament d'excepcions i creant un *log* per controlar l'execució d'aquests.



Documentació dels procediments implementats.

Creació de jocs de prova per realitzar el test del funcionament de la base de dades.

Realització de la memòria del Treball Final de Carrera.

Realització presentació Treball Final de Carrera

### 1.5 Eines a utilitzar

*OpenProject* per realitzar el control de la planificació del projecte.

*OpenOffice* i/o *Microsoft Office* per la redacció de la memòria.

PowerPoint per la realització de la presentació del TFC

L'aplicació *Dia* per realitzar el diagrama E/R.

*Oracle Express v.10.2.0.1* com a Sistema de Gestió de BBDD.

*Sql Developer v1.0.0.15* com IDE per treballar, generar i executar *scripts* de sql amb la BD.

*Microsoft Visio* com a eina de suport complementari per la realització de diagrames.

### 1.6 Planificació del projecte

Nombre	Duración	Inicio	Terminado
Disseny E/R	3 days	21/03/11 0:00	24/03/11 0:00
Script Sql alumnes i professors	3 days	25/03/11 8:00	28/03/11 8:00
Script Sql cursos i assignatures	3 days	28/03/11 8:00	31/03/11 8:00
Script Sql calendari escolar	3 days	31/03/11 8:00	3/04/11 8:00
Script Tipus amonestacions i sancions	3 days	3/04/11 8:00	6/04/11 8:00
Script gestió monestacions i sancions	7 days	6/04/11 8:00	13/04/11 8:00
Procediments ABM alumnes	3 days	16/04/11 8:00	19/04/11 8:00
Procediments ABM professors, cursos i assignatures	3 days	19/04/11 8:00	22/04/11 8:00
Procediments ABM calendari i tipus amonest. i sancions	3 days	22/04/11 8:00	25/04/11 8:00
Procediments consultes	7 days	29/04/11 8:00	6/05/11 8:00
Procediments mòdul estadístic	7 days	6/05/11 8:00	13/05/11 8:00
Documentació creació taules i triggers	22 days	25/03/11 8:00	16/04/11 8:00
Documentació procediments	24 days	16/04/11 8:00	10/05/11 8:00
Jocs de proves i Test	14 days	14/05/11 8:00	28/05/11 8:00
Memoria projecte	80 days	23/03/11 8:00	11/06/11 8:00
Presentació projecte	14 days	28/05/11 8:00	11/06/11 8:00
PAC2	28 days	21/03/11 0:00	18/04/11 0:00
PAC3	28 days	17/04/11 23:59	15/05/11 23:59
LLIURAMENT TFC	83 days	21/03/11 0:00	12/06/11 0:00



L'objectiu marcat per la fase1 o pac2 no és arriscat, i per tant, el seu compliment es pot avançar. Tot el contrari del compliment de la fase 2 o pac3, el qual pot veure's afectat, per la complicació, a l'hora de realitzar segons quins procediments, a més de la realització de les proves pertinents.

La finalització avançada de la pac2 permetrà l'avançament del inici del desenvolupament dels procediments, i per tant, minimitzar els riscos de retràs dins aquesta fase.

La realització del joc de proves, finalitzar la memòria i la presentació del projecte, englobat tot a la fase final del projecte, dins la qual, no es preveu retràs en la seva confecció a menys que es dugui retràs en les fases anteriors.

El possible retràs a la fase 2 o pac3, i per tant, la segura complicació a la fase final, provocarà ampliar l'horari de feina en una hora diària.

Com a previsió es realitzarà comprovacions de feina els divendres de cada setmana, des de la primera fase (pac 2), i en cas de detectar-ne un possible retràs, s'habilitaran hores complementaries de feina els dissabte i diumenge següents per començar el dilluns segons l'agenda definida.

## **1.7 Productes Obtinguts**

A la finalització del projecte, es lliurarà:

- Memòria del TFC de Base de Dades Relacionals corresponent al disseny i implementació d'un Sistema de gestió d'amonestacions i sancions en centres educatius.
- Script de creació de la base de dades, a on s'inclou la creació de les taules, índexs, disparadors i procediments.
- Documentació per l'execució del script i breu explicació del seu contingut.
- Joc de proves utilitzat per realitzar el test del funcionament.
- Documentació del test d'estrès aplicat a la base de dades.
- Document presentació del TFC

## **2. Desenvolupament BD**

### **2.1 Introducció**

Els requeriments del disseny de la base de dades per gestionar el sistema de gestió d'amonestacions i sancions en centres educatius, són els següents:

- Crear la persistència que s'utilitzarà a cada centre educatiu.
- Gestionar les dades dels professors de cada centre.
- Gestionar les dades dels alumnes del centre.
- Configurar el calendari escolar del centre.
- El centre escolar imparteix els cursos corresponents a cada etapa escolar.
- Cada curs es compon d'assignatures, les quals amb la mateixa nomenclatura només s'imparteixen a un curs concret.
- Mantenir un control de horaris de tutories.
- La gestió de amonestacions i sancions ha de permetre:
  - Gestionar tipus d'amonestacions diferents
  - Crear regles de sancions segons la reiteració d'amonestacions pel seu tipus.
  - Un tipus d'amonestació concret només pot generar un tipus de sanció concreta.
  - Es parteix d'uns tipus inicials predefinits i d'unes regles ja predefinides, encara que el sistema ha de permetre la seva modificació i actualització.
  - Permetre dur un registre de les sancions realitzades als alumnes.



## 2.2.2. Atributs de les entitats

### **PERSONA**

idPersona, documentIdentitat, tipusDocument, nom, primerCognom, segonCognom, domicili, codiPostal, poblacio, idPais, idNacionalitat, dataNaixment, telefon, movil, email, observacio

### **ALUMNE (subclasse de PERSONA)**

idPersona

### **PROFESSOR (subclasse de PERSONA)**

idPersona

### **PAIS**

idPais, nomPais, descripcioNacionalitat

### **TITULACIO**

idTitulacio, nomTitulacio

### **CURS**

idCurs, nomCurs

### **ASSIGNATURA**

idAssignatura, nomAssignatura

### **PERIODE**

idPeriode, dataInici, dataFinal

### **FESTA**

Dia, motiu

### **SEMANALECTIVA**

idSemana, dilluns, dimarts, dimecres, dijous, divendres, dissabte, diumenge

### **SALA**

idSala, situacio, aforo

## **AULA**

idAula

## **MATRICULA**

idMatricula

## **HORARI**

idHorari, descripcio

## **HORARIDIA**

idHorariDia, diaSemana, horaInicial, horaFinal

## **TIPUSAMONESTACIO**

idTipusAmonestacio, descripcio, nivellGravetat

## **TIPUSSANCIO**

idTipusSancio, descripcio

## **AMONESTACIO**

idAmonestacio data, hora, comunicacio, observacio

## **SANCIO**

idSancio, dataSancio, motiu, comunicacio, observacio

## **DATA**

data

## **HORA**

hora

### **2.2.3. Justificació de la proposició realitzada**

#### **2.2.3.1. Generalitzacions / Especialitzacions**

S'ha definit una generalització de les entitats ALUMNE i PROFESSOR amb la de PERSONA, per incloure tots els atributs comuns. Aquesta generalització serà total i disjunta, ja que tota persona definida dins l'aplicació ha de ser o ALUMNE o PROFESSOR, i no es contempla que un ALUMNES pugui ser professor ni a l'inversa.

### **2.2.3.2. Entitats associatives**

S'ha definit CURS ESCOLAR com una interrelació M:N, que defineix per cada curs un període escolar. CURS ESCOLAR es considera entitat associativa, amb l'atribut idCursEscolar, per poder-la interrelacionar amb SEMANALECTIVA, amb AULA i amb MATRICULA, interrelació per cada una de les quals serà explicada al punt 2.2.3.3 d'aquest document.

S'ha definit AMONESTACIO com una interrelació ternària M:N:P, que recull les amonestacions, de tots els tipus, fetes pels professors als alumnes. AMONESTACIO és una entitat associativa per poder interrelacionar amb SANCIO quan l'amonestació sigui greu o hagi un acumulat de amonestacions segons unes regles dictades.

### **2.2.3.3. Interrelacions, connectivitat i dependència**

VIU → 1:N

La interrelació VIU defineix el país on té el seu domicili habitual una persona. Lògicament, hi ha N persones que viuen a un mateix país, però en canvi, una persona només té un domicili habitual.

Poden existir països donats d'alta, però que no tenen cap persona enregistrada, per tant, no és obligatòria en aquest sentit.

NACIONALITAT → 1:N

La interrelació NACIONALITAT defineix el país de la nacionalitat de la persona. . Lògicament, hi ha N persones que tenen una mateixa nacionalitat, però en canvi, una persona només té una nacionalitat (els casos de doble nacionalitat no es té en compte).

Poden existir països donats d'alta, però que no tenen cap persona que tinguin aquesta nacionalitat, per tant, no és obligatòria en aquest sentit.

INSCRIT → 1:N

La interrelació INSCRIT defineix la inscripció d'un alumne mitjançant una matricula. Un alumne pot tenir tantes matricules com cursos escolar estigui o hagi cursat, però només una matricula pot pertany a un alumne.

Un alumne es dona d'alta amb la primera matricula i una matricula no pot existir sense un alumne, per tant, la interrelació és obligatòria en els dos sentits.

FORMAT PER → 1:N



La interrelació FORMAT PER, defineix que una titulació està definida per un numero indeterminat de cursos. A més, un curs només pertany a una titulació. L'obligatorietat és òbvia en ambdós sentits.

TE → 1:N

La interrelació TE, defineix les assignatures que componen un CURS. Una assignatura només por pertany a un CURS. L'obligatorietat és òbvia en ambdós sentits.

AVALUACIO → 1:M:N

La interrelació AVALUACIO, defineix l'avaluació que realitza cada professor de les assignatures amb els alumnes matriculats a aquestes. Una avaluació d'una assignatura per una matricula només té un professor, en canvi, un professor pot avaluar una matricula més d'una assignatura (N), i a més, un professor avalua una assignatura per varies matricules (M). En totes elles, sempre hi haurà un atribut Nota per aquesta relació.

CURSESCOLAR → N:M

La interrelació CURSESCOLAR, defineix es períodes escolars en que un curs es dona docència. Un curs es dona docència a M períodes i dins un període es dona docència a N cursos.

Un curs no té sentit sense períodes de docència, i un període no té sentit definir-lo si no és per cursar docència.

NOLECTIUS → 1:N

La interrelació NOLECTIUS, defineix l'assignació dels dies festius a un període donat d'alta. Una festa només correspon a un període, en canvi, un període pot tenir N festes.

Una festa s'ha de assignar obligatòriament com a no lectiu a un període, en canvi, un període, pot no tenir cap festa.

DIES LECTIUS → 1:N

La interrelació DIES LACTIUS, defineix la configuració de la setmana lectiva per un curs escolar. Per cada curs escolar és defineix una sola configuració, no es contempla el canvi a mitjan curs del canvi de dies no lectius setmanals. Una configuració de dies lectius setmanals pot repetir-se a N cursos escolars.

Aquesta interrelació és obligatòria en ambdós sentits, ja que si al curs escolar no se li defineix la setmana lectiva no es podrien configurar horaris, i una configuració de setmana lectiva no té sentit crear-la si no és per assignar-la a un curs escolar.

IMPARTEIX → 1:N

La interrelació IMPARTEIX, defineix les matricules que tenen docència a un curs escolar. Cada matrícula només correspon a un curs escolar, i lògicament a un curs escolar pot haver N matricules.

Un curs escolar sense matricules o una matrícula sense docència no té sentit, per tant, la interrelació és obligatòria en ambdós sentits.

AGRUPA → 1:N

La interrelació AGRUPA, defineix la formació de grups o aules dins un curs escolar on s'impartirà la docència. Un curs escolar té N aules, en canvi una aula pertany només a un curs escolar.

Un curs escolar sense aules no té sentit, ni una aula sense pertany a un curs escolar. Això implica obligatorietat en ambdós sentits.

UBICAT → 1:1

La interrelació UBICAT, defineix la sala que utilitzarà un aula. Una sala només es preveu que sigui utilitzada per un aula, i una aula només utilitzarà una mateixa sala.

Una aula sense espai físic no té sentit, i per tant, la interrelació és obligatòria en aquest sentit. En canvi, poden haver sales no destinades a docència, i per tant, no tenen aula assignada.

ASSIGNACIO → 1:N

La interrelació ASSIGNACIO, defineix la composicions de les aules dels alumnes matriculats. Una matrícula només pot ser assignada a un aula, en canvi, un aula pot tenir N matricules.

Una matrícula inscrita, s'ha d'assignar obligatòriament a un aula, i un aula no té sentit sense matricules assignades, per tant, la interrelació és obligatòria en ambdós sentits.

TUTORIA → 1:N

La interrelació TUTORIA, defineix la assignació d'un professor com a tutor a un aula. Un professor pot ser tutor de varies aules, però un aula només pot tenir un tutor.

Un aula ha de tenir un tutor obligatòriament, però un professor no té perquè ser tutor d'un aula.

HORARI TUTORIA → 1:N

La interrelació HORARI TUTORIA, defineix la definició del horari de tutoria a una aula. Una aula només pot tenir un horari de tutoria, encara que un mateix horari pot servir per varies aules com a horari de tutoria.

Una aula obligatòriament, ha de tenir un horari de tutoria. Però un horari definit, pot ser hagi estat definit per horari de classes, i no de tutories, i per tant, no té obligatorietat.

#### HORARI SEMANA → 1:N

La interrelació HORARI SEMANA, defineix un horari amb els dies i hores que el componen. Si l'entitat HORARIDIA defineix un dia de la setmana amb un interval d'hores determinades, l'entitat HORARI, pot tenir més d'un dia de la setmana assignat, fins i tot, el mateix dia es pot repetir en diferents dies. Un HORARIDIA pot estar assignat a més d'un horari.

Com que un horari sense dies configurats no té sentit, i la configuració de dies sense assignar-ho a un horari tampoc, la interrelació és obligatòria en ambdós sentits.

#### CLASSE → 1:1:1

La interrelació CLASSE, defineix l'assignació d'un horari per cada assignatura a cada aula. Una aula no es comparteix, es a dir, dins una aula, només pot haver una assignatura per un horari, i lògicament, només a un horari s'impartirà una assignatura. Una assignatura, no es preveu, que s'imparteixi dins al mateix horari a aules diferents.

#### AMONESTACIO → N:M:P:R:S

La interrelació AMONESTACIO, relaciona les amonestacions fetes als alumnes per professors determinant el tipus d'amonestació que és i a una data i hora determinada. Un alumne pot rebre d'un professor més d'una amonestació del mateix tipus. Un professor pot fer una amonestació del mateix tipus a un alumne. I un tipus d'amonestació pot repetir-se a diferents amonestacions amb indiferència del alumne i professor.

#### REGLES SANCIO → N:M

La interrelació REGLES SANCIO, indica en quin cas un tipus d'amonestacions pot rebre un tipus de sanció. Un tipus d'amonestació segons la seva repetició pot tenir una sanció. Una sanció pot aplicar-se per la repeticions d'una sèrie d'amonestacions.

Un tipus d'amonestació ha de tenir una sanció configurada i una sanció ha de provenir sempre d'una amonestació prèvia. Per tant, la interrelació és obligatòria en ambdós sentits.

#### GREU O ACUMULA → 1:N

La interrelació GREU O ACUMULA, indica la creació d'una sanció debut a amonestacions greus o acumulacions de lleus. Una sanció correspon a una o varies amonestacions i una amonestació concreta només pot implicar una sanció.

Una amonestació, si és lleu, pot ser que mai provoqui amonestació, per tant, no és obligatòria la interrelació en aquest sentit. Al contrari passa a l'inrevés, una sanció obligatòriament prové d'una o varies amonestacions.

DE TIPUS → 1:N

La interrelació DE TIPUS, indica el tipus de sanció d'una sanció imposada. Un conjunt de sancions poden tenir el mateix tipus, però per cada una, només són d'un tipus.

Poden existir tipus de sancions, que mai hagin estat sancionades, per tant, no és obligatòria la interrelació en aquest sentit. Al contrari passa a l'inrevés, una sanció obligatòriament és d'un tipus de sanció.

## 2.3 Disseny lògic BD

### 2.3.1. Entitats

#### PERSONA

(idPersona, documentIdentitat, tipusDocument, nom, primerCognom, segonCognom, domicili, codiPostal, poblacio, idPais, idNacionalitat, dataNaixment, telefon, movil, email, observacio)

{idPais} és clau forana de PAIS(idPais)

{idNacionalitat} és clau forana de PAIS(idPais)

#### ALUMNE

(idPersona )

Entitat subclasse de PERSONA

#### PROFESSOR

( idPersona )

Entitat subclasse de PERSONA

#### PAIS

(idPais, nomPais, descripcioNacionalitat)

#### TITULACIO

(idTitulacio, nomTitulacio)

#### CURS

(idCurs, nomCurs, idTitulacio)

{idTitulacio} és clau forana de TITULACIO(idTitulacio)

#### ASSIGNATURA

(idAssignatura, nomAssignatura, idCurs )

{idCurs} és clau forana de CURS(idCurs)

## **PERIODE**

(idPeriode, dataInici, dataFinal )

## **FESTA**

(dia, motiu, idPeriode)

{idPeriode} és clau forana de PERIODE(idPeriode)

## **SEMANALECTIVA**

(idSemana, dilluns, dimarts, dimecres, dijous, divendres, dissabte, diumenge)

## **SALA**

(idSala, situacio, aforo)

## **AULA**

(idAula, idPeriode, idCurs, idSala, idTutor, idHorariTutoria)

{idPeriode, idCurs} és clau forana de CURSESCOLAR(idPeriode, idCurs)

{idPeriode} és clau forana de PERIODE(idPeriode)

{idCurs} és clau forana de CURS(idCurs)

{idSala} és clau forana de SALA(idSala)

{idTutor} és clau forana de PROFESSOR(idPersona)

{idHorariTutoria} és clau forana de HORARI(idHorari)

## **MATRICULA**

(idMatricula, idAlumne, idPeriode, idCurs, idAula)

{idAlumne} és clau forana de ALUMNE(idPersona)

{idPeriode, idCurs} és clau forana de CURSESCOLAR(idPeriode, idCurs)

{idPeriode} és clau forana de PERIODE(idPeriode)

{idCurs} és clau forana de CURS(idCurs)

{idAula} és clau forana de AULA(idAula)

## **HORARI**

(idHorari, descripcio )

## **HORARIDIA**

(idHorariDia, diaSemana, horaInicial, horaFinal )

### **TIPUSAMONESTACIO**

(idTipusAmonestacio, descripcio, nivellGravetat, idTipusSancio )

{idTipusSancio} és clau forana de TIPUSSANCIO(idTipusSancio)

### **TIPUSSANCIO**

(idTipusSancio, descripcio )

### **SANCIO**

(idSancio, dataSancio, motiu, comunicacio, observacio, idTipusSancio)

{idTipusSancio} és clau forana de TIPUSSANCIO(idTipusSancio)

## **2.3.2. Interrelacions**

### **CURSESCOLAR**

(idPeriode, idCurs, idSemana)

{idPeriode} és clau forana de PERIODE(idPeriode)

{idCurs} és clau forana de CURS(idCurs)

{idSemana} és clau forana de SEMANALECTIVA(idSemana)

### **CLASSE**

(idAula, idAssignatura, idHorari )

{idAula} és clau forana de AULA(idAula)

{idAssignatura} és clau forana de ASSIGNATURA(idAssignatura)

{idHorari} és clau forana de HORARI(idHorari)

### **AVALUACIO**

(idAssignatura, idProfessor, idMatricula, nota, observacio )

{idAssignatura} és clau forana de ASSIGNATURA(idAssignatura)

{idProfessor} és clau forana de PROFESSOR(idPersona)

{idMatricula} és clau forana de MATRICULA(idMatricula)

### **HORARISEMANA**

(idHorari, idHorariDia )

{idHorari} és clau forana de HORARI(idHorari)

{idHorariDia} és clau forana de HORARIDIA(idHorariDia)

## **AMONESTACIO**

(idAlumne, idProfessor, idTipusAmonestacio, data, hora, comunicacio, observacio, idSancio )

{idAlumne} és clau forana de ALUMNE(idPersona)

{idProfessor} és clau forana de PROFESSOR(idPersona)

{idSancio} és clau forana de SANCIO(idSancio)

{idTipusAmonestacio} és clau forana de

TIPUSAMONESTACIO (idTipusAmonestacio)

## **2.4 Scripts BD**

### **2.4.1. Ajusts del disseny**

Debut a que el Sistema de Gestió de Base de Dades utilitzat és l'Oracle Express, i que aquest sistema no permet la implantació de la herència de taules, la generalització definida de ALUMNES i PROFESSOR definit al model queda eliminat quedant les entitats de la següent forma:

#### **ALUMNE**

(idAlumne, documentIdentitat, tipusDocument, nom, primerCognom, segonCognom, domicili, codiPostal, poblacio, idPais, idNacionalitat, dataNaixment, telefon, movil, email, observacio)

{idPais} és clau forana de PAIS(idPais)

{idNacionalitat} és clau forana de PAIS(idPais)

#### **PROFESSOR**

(idProfessor, documentIdentitat, tipusDocument, nom, primerCognom, segonCognom, domicili, codiPostal, poblacio, idPais, idNacionalitat, dataNaixment, telefon, movil, email, observacio)

{idPais} és clau forana de PAIS(idPais)

{idNacionalitat} és clau forana de PAIS(idPais)

Les entitats i relacions referenciades a aquestes entitats també han estat canviades per els identificadors dins les claus foranes.

#### **AULA**

(idAula, idPeriode, idCurs, idSala, idTutor, idHorariTutoria)

{idPeriode, idCurs} és clau forana de CURSESCOLAR(idPeriode, idCurs)

{idPeriode} és clau forana de PERIODE(idPeriode)

{idCurs} és clau forana de CURS(idCurs)

{idSala} és clau forana de SALA(idSala)

{idTutor} és clau forana de PROFESSOR(idProfessor)

{idHorariTutoria} és clau forana de HORARI(idHorari)

### **MATRICULA**

(idMatricula, idAlumne, idPeriode, idCurs, idAula)

{idAlumne} és clau forana de ALUMNE(idAlumne)

{idPeriode, idCurs} és clau forana de CURSESCOLAR(idPeriode, idCurs)

{idPeriode} és clau forana de PERIODE(idPeriode)

{idCurs} és clau forana de CURS(idCurs)

{idAula} és clau forana de AULA(idAula)

### **AVALUACIO**

(idAssignatura, idProfessor, idMatricula, nota, observacio )

{idAssignatura} és clau forana de ASSIGNATURA(idAssignatura)

{idProfessor} és clau forana de PROFESSOR(idProfessor)

{idMatricula} és clau forana de MATRICULA(idMatricula)

### **AMONESTACIO**

(idAlumne, idProfessor, idTipusAmonestacio, data, hora, comunicacio, observacio, idSancio )

{idAlumne} és clau forana de ALUMNE(idAlumne)

{idProfessor} és clau forana de PROFESSOR(idProfessor)

{idSancio} és clau forana de SANCIO(idSancio)

{idTipusAmonestacio} és clau forana de

TIPUSAMONESTACIO (idTipusAmonestacio)

Altre punt a tenir en compte és el tractament de les hores al Sistema de Gestió de Base de Dades Oracle Express, ja que està inclòs dins el tipus de dades DATA. Això afecta a la relació AMONESTACIO, la qual unifica els atributs DATA i HORA en un mateix.

### **AMONESTACIO**

(idAlumne, idProfessor, idTipusAmonestacio, data, comunicacio, observacio, idSancio )

{idAlumne} és clau forana de ALUMNE(idAlumne)

{idProfessor} és clau forana de PROFESSOR(idProfessor)

{idSancio} és clau forana de SANCIO(idSancio)

{idTipusAmonestacio} és clau forana de

TIPUSAMONESTACIO (idTipusAmonestacio)



### 3. Implementació BD

#### 3.1 Introducció

Dins la fase d'implementació, es detallen els procediments operatius de la base de dades, necessaris per gestionar el funcionament del sistema de gestió d'amonestacions i sancions dins els centres d'educació.

Per cada requisit operatiu, es detallarà el seu procediment emmagatzemat associat amb la funcionalitat d'aquest, una breu explicació del seu funcionament, i la seva sintaxis d'execució amb l'explicació dels seus paràmetres d'execució i de retorn, anomenats també paràmetres d'entrada i de sortida.

#### 3.2 Taules de persistència per l'estadística

El mòdul estadístic, ha de donar solucions a les consultes alimentades per la gestió d'amonestació i gestió, segons es detalla al plec tècnic.

Per realitzar aquest mòdul, i per complir el requisit d'oferir les dades en temps constant 1, s'han creat per cada una de les consultes, una taula de persistència, la qual serà alimentada dins l'execució dels procediments definits, pel propi procés de gestió de amonestacions i sancions, i amb una sèrie de procediments creats a tal efecte.

---

<b>Nom taula</b>	<b>Descripció de la consulta</b>
<b>E1</b>	Número d'amonestacions per alumne
<b>E2</b>	Número de sancions per alumne i any
<b>E3</b>	Mitjana d'amonestacions per professor i any
<b>E4</b>	El número de sancions per curs i any
<b>E5</b>	El nom de l'alumne més sancionat en un any donat
<b>E6</b>	El nom del professor més amonestador per curs
<b>E7</b>	La mitjana de sancions que tenen els alumnes per curs
<b>E8</b>	El número d'alumnes que no tenen cap amonestació

---

### 3.3 Logs

Debut a requisists de la pròpia metodologia del sistema, es crearà una taula per enregistrar totes les cridades als procediments que a continuació es definiran.

Aquesta taula es denominarà log i enregistrarà el nom de procediment executat, els paràmetres d'entrada i el de sortida.

### 3.4 Procediments

#### 3.4.1. Introducció als procediments

Els requeriments funcionals del sistema dona solució als requisits del plec tècnic, i per fer-ho s'han creat una sèrie de procediments, els quals afecten a cada una de les taules creades.

Als procediments cobreixen quatre finalitats, ben definides dins el plec tècnic:

a) Manteniment de les taules de persistència

Totes les taules tenen els procediments que permeten controlar la inserció, eliminació i modificació de les dades realitzant la comprovació de la integritat i l'enregistrament, dins la taula log, del resultat de l'execució del procediment.

Tots aquests procediments, contenen la mateixa estructura amb independència de la taula de persistència a la qual fa referència.

Com a cas excepcional, debut a la composició reduïda d'atributs, la taula *CursEscolar* no té procediment de modificació de dades, només de inserció i de eliminació.

b) Control de sancions

Per la assignació automatitzada de sancions, s'han creat dos procediments que s'executen quan es realitza una nova amonestació.

c) Llistat de dades bàsiques

S'han creat els procediments necessaris per la consulta de les dades bàsiques segons els requeriments del plec de condicions.

Aquestes son:

- Llistat amonestacions imposades
- Llistat d'alumnes d'un curs
- Llistat de tipus amonestacions i sancions
- Llistat de les amonestacions i sancions d'un alumne

d) Mòdul estadístic

S'han creat els procediments per alimentar les taules de persistència del mòdul estadístic. Aquests procediments són cridats des de els procediments d'alta, modificació i eliminació de les amonestacions i sancions.

### 3.4.2. Esquema de procediments

A continuació es detalla l'esquema general dels procediments.

#### 3.4.2.1. Manteniment de taules de persistència

##### 3.4.2.1.1. Procediment per la inserció de dades

- Execució del procediment

La sintaxi de l'execució del procediment és:

Nom\_procediment  
( paràmetre entrada 1, ..., paràmetre entrada n,  
 [paràmetre sortida <id>,) ]  
 paràmetre sortida <resultat> )

- Nom del procediment

Els procediments de inserció de dades tenen el nom format per la preposició "p\_", seguit del nom de la taula de persistència afectada, i acaben amb la terminació "\_add".

Per exemple: *p\_alumne\_add* és el nom del procediment d'inserció dins la taula *alumne*.

- Paràmetres d'entrada

Els paràmetres d'entrada que tenen els procediments d'insercions, són cada un dels atributs de la taula, exceptuant l'atribut identificador en els casos de que aquest sigui un seqüencial.

Exemple : Per l'inserció a la taula Alumne els paràmetres serien:

Document d'identitat del alumne, tipus de document d'identitat, nom del alumne, primer cognom, segon cognom, domicili, codi postal del domicili, població del domicili, codi del país enregistrat a la taula *país* del domicili, codi del país enregistrat a la taula *país* de la nacionalitat del alumne, data naixement, telèfon, mòbil, observacions

- Paràmetre de sortida <id>

Per els casos de taules on l'identificador és un seqüència, el paràmetre de sortida <id> retorna el id que ha donat d'alta el Sistema de Gestió de Base de Dades, per el registre inserit.

Per els casos de taules on no hi ha un identificador en format de seqüència (per exemple, cursEscolar) aquest paràmetre de sortida no figurarà a l'execució del procediment.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**).

- Anàlisi funcional del procediment.

Tots els procediments d'inserció tenen la mateixa estructura funcional. Aquesta es subdivideix en :

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la inserció.
- Realitza la inserció de la taula
- Si la inserció ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

- Cas especial 1.

La inserció a la taula *amonestacio*, és un cas especial, debut el control que s'ha de fer per l'assignació automàtica de sancions i la generació de les dades pel mòdul estadístic.

Per aquest cas, concretament el procediment "*p\_amonestacio\_add*", el anàlisi funcional és:

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la inserció.
- Realitza la inserció de la taula
- Si la inserció ha estat correcta executa el procediment *p\_amonestacio\_control*, el qual comprovarà que, amb la nova amonestació, l'alumne no hagués de ser sancionat segons els criteris establerts. Si aquest procediment de control reporta qualque excepció

aquesta també serà reportada com error mitjançant el procediment d'inserció.

- Si l'execució del procediment de control ha estat correcte, realitza l'execució dels procediments per generar les estadístiques, concretament les que afecten a les taules E1, E3, E6 i E8. Els errors que aquests reportin també seran reportats.
- Si la comprovació de control no ha reportat cap error, s'insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

- Cas especial 2.

La inserció a la taula *sanció*, també és un pot utilitzat per alimentar de dades el mòdul estadístic.

Per aquest cas, concretament el procediment "*p\_sancio\_add*", el anàlisi funcional és:

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la inserció.
- Realitza la inserció de la taula
- Realitza l'execució dels procediments per generar les estadístiques, concretament les que afecten a les taules E2, E4 i E7. Els errors que aquests reportin també seran reportats.
- Si la comprovació de control no ha reportat cap error, s'insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### **3.4.2.1.2. Procediment per la modificació de dades**

- Execució del procediment

La sintaxi de l'execució del procediment és:

Nom\_procediment  
( paràmetre entrada 1, ..., paràmetre entrada n,  
paràmetre sortida <resultat> )

- Nom del procediment

Els procediments de modificació de dades tenen el nom format per la preposició "p\_", seguit del nom de la taula de persistència afectada, i acaben amb la terminació "\_update".

Per exemple: *p\_alumne\_update* és el nom del procediment que permet la modificació un registre dins la taula *alumne*.

- Paràmetres d'entrada

Els paràmetres d'entrada que tenen els procediments de modificacions, són cada un dels atributs de la taula.

Exemple : Per modificar la taula Alumne els paràmetres serien:

Identificador del registre del alumne, document d'identitat del alumne, tipus de document d'identitat, nom del alumne, primer cognom, segon cognom, domicili, codi postal del domicili, població del domicili, codi del país enregistrat a la taula *país* del domicili, codi del país enregistrat a la taula *país* de la nacionalitat del alumne, data naixement, telèfon, mòbil, observacions

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.

Tots els procediments de modificació tenen la mateixa estructura funcional. Aquesta es subdivideix en :

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
També comprova que l'identificador passat com a primer paràmetre existeix dins la taula.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la modificació.
- Realitza la modificació de la taula
- Si la modificació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

- Cas especial 1.

La modificació de la taula sanció, és utilitzada per actualitzar les dades del mòdul estadístic, per tant, el seu anàlisi funcional té una petita variació.

Per aquest cas, concretament el procediment "p\_sancio\_update", el anàlisi funcional és:

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
També comprova que l'identificador passat com a primer paràmetre existeix dins la taula.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la modificació.
- Realitza la modificació de la taula
- Realitza l'execució dels procediments, en cas de canviar l'alumne o la data de sanció, per desactualitzar les dades anteriors i actualitzar les noves, i així, regenerar les estadístiques, concretament les que afecten a les taules E2, E4 i E7. Els errors que aquests reportin també seran reportats.
- Si la modificació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment

(El procediment p\_amonestacio\_update no permet al modificació de dades que puguin alterar l'estadística).

### 3.4.2.1.3. Procediment per la eliminació de dades

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
Nom_procedimient
( paràmetre entrada <id1>,
  [paràmetre entrada <id2>, ..., paràmetre entrada <idn>, ]
  paràmetre sortida <resultat>)
```

- Nom del procediment

Els procediments de eliminació de dades tenen el nom format per la preposició "p\_", seguit del nom de la taula de persistència afectada, i acaben amb la terminació "\_delete".

Per exemple: *p\_alumne\_delete* és el nom del procediment que permet l'eliminació d'un registre dins la taula *alumne*.

- Paràmetres d'entrada

Els paràmetres d'entrada que tenen els procediments de eliminació, són els identificadors o clau primàries de la taula. Normalment, només hi haurà un paràmetre d'entrada, ja que, la majoria de les taules té un camp identificador únic.

Exemple :

Per eliminar un registre de la taula Alumne el paràmetre seria :

Identificador del registre del alumne

Per eliminar un registre de la taula Amonestacio els paràmetres serien:

Els identificadors del alumne, del professor i del tipus d'amonestació, a més de la data de la amonestació.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.

Tots els procediments de eliminació tenen la mateixa estructura funcional. Aquesta es subdivideix en :

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent. També comprova, que el registre a eliminar existeix. En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la eliminació.
- Realitza la eliminació de la taula
- Si la eliminació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.



- Cas especial 1.

La eliminació a la taula *amonestacio*, també alimenta les taules que afecten al mòdul estadístic.

Per aquest cas, concretament el procediment "*p\_amonestacio\_delete*", el anàlisi funcional és:

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
També comprova, que el registre a eliminar existeix.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la eliminació.
- Realitza la eliminació de la taula
- Realitza l'execució dels procediments per generar les estadístiques, concretament les que afecten a les taules E1, E3, E6 i E8. Els errors que aquests reportin també seran reportats
- Si la eliminació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

- Cas especial 2.

La eliminació a la taula *sancio*, també alimenta les taules que afecten al mòdul estadístic.

Per aquest cas, concretament el procediment "*p\_sancio\_delete*", el anàlisi funcional és:

- Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
- Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors i, si són identificadors d'un altre taula, comprova la seva integritat dins la seva taula corresponent.  
També comprova, que el registre a eliminar existeix.  
En cas de qualque incidència la reporta mitjançant una excepció identificativa de la mateixa i no realitza la eliminació.
- Realitza la eliminació de la taula
- Realitza l'execució dels procediments per generar les estadístiques, concretament les que afecten a les taules E2, E4 i E7. Els errors que aquests reportin també seran reportats
- Si la eliminació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### 3.4.2.2. Control de sancions

#### 3.4.2.2.1. Procediment assignació automàtica de sancions

Procediment que comprova si s'ha de imposar qualche sanció amb les amonestacions realitzades.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_amonestacio_control  
( paràmetre entrada <idAlumne>,  
  paràmetre entrada <idTipusAmonestació>,  
  paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne amonestat.

<idTipusAmonestació>  
L'identificador del tipus de amonestació darrer que ha tengut l'alumne.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors.  
Com els paràmetres d'entrada han estat reportats des de un altre procediment no es realitza el control d'integritat.
  - Cerca el número d'amonestacions que corresponen els identificadors del alumne i tipus d'amonestació que no tinguin cap sanció.
  - Cerca el nivell de gravetat del tipus d'amonestació, així com el tipus de sanció aplicar si es compleix.
  - Si s'ha arribat al nivell de gravetat per part de l'alumne es realitzaran les següents accions.
    - Bloquejar la taula sanció.
    - Donar d'alta la sanció a l'alumne mitjançant la cridada al procediment p\_sancio\_add

- Si aquest procediment retorna Ok, es a dir, la sanció ha estat correctament inserida, es realitzen les següents accions:
  - Marcar les amonestacions del alumne que han motivat la sanció amb el identificador de la sanció creada, mitjançant la crida al procediment p\_amonestacio\_sancio
  - Realitzar la inserció del log per indicar que el procediment ha estat correcta.
 Si no ha estat ben inserida es desfarà l'operació d'inserció de la sanció i es retornarà la incidència corresponent.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.2.2. Procediment actualitzar amonestacio-sancio.**

Procediment que actualitza una amonestació amb la sanció que ha generat.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_amonestacio_sancio
( paràmetre entrada <idAlumne>,
  paràmetre entrada <idProfessor>,
  paràmetre entrada <idTipusAmonestació>,
  paràmetre entrada <data>,
  paràmetre entrada <idTipusSancio>,
  paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne amonestat.

<idProfessor>  
L'identificador del professor que ha realitzat la amonestació.

<idTipusAmonestació>  
L'identificador del tipus de amonestació darrer que ha tengut l'alumne.

<data>  
Data de l'amonestació

<idTipusSancio>  
L'identificador del tipus de sanció que ha generat l'amonestació.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.  
Comprova que els paràmetres obligatoris tinguin valors.
  - Actualitza l'amonestació amb l'identificador de la sanció.
  - Si la modificació ha estat correcta insereix el log corresponent i retorna OK com a paràmetre de sortida.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### **3.4.2.3. Llistats dades bàsiques**

#### **3.4.2.3.1. Llistat dades bàsiques d'un alumne**

Procediment que llista les dades bàsiques d'un alumne.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_alumne
  ( paràmetre entrada <idAlumne>,
    paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Si l'alumne existeix obté les dades bàsiques d'aquest, si no reporta la incidència com excepció.
  - Realitza la impressió del nom, el dni, domicili, població email i telèfons del alumne.
  - El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### 3.4.2.3.2. Llistat alumnes d'un curs

Procediment que llista els alumnes matriculat per un període escolar per un curs determinat.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_alumnes_curs
  ( paràmetre entrada <idPeriode>,
    paràmetre entrada <idAlumne>,
    paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idPeriode>  
L'identificador del període escolar.

<idAlumne>  
L'identificador del alumne.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prepara, dins la definició, el cursor amb els alumnes matriculats segons els paràmetres.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.

- Comprova que els paràmetres d'entrada tinguin valors i siguin correctes.
- Realitza la impressió dels alumnes matriculats, que reporten el cursor preparat, executant el procediment p\_llistat\_alumne.
- El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.3.3. Llistat tipus d'amonestacions**

Procediment que llista els tipus d'amonestacions donats d'alta.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_tipusA
( paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

No té paràmetres d'entrada.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments**)..

- Anàlisi funcional del procediment.
  - Prepara, dins la definició, el cursor amb els tipus d'amonestacions.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Realitza la impressió dels tipus d'amonestacions que reporta el cursor preparat.
  - El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.3.4. Llistat tipus de sancions**

Procediment que llista els tipus de sancions donats d'alta.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_tipusSancio  
( paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

No té paràmetres d'entrada.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prepara, dins la definició, el cursor amb els tipus de sanció.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Realitza la impressió dels tipus de sanció que reporta el cursor preparat.
  - El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.3.5. Llistat de amonestacions d'un alumne**

Procediment que llista les amonestacions d'un alumne.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_sancio_alumne  
( paràmetre entrada <idAlumne>,  
  paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>

L'identificador del alumne.

Pot tenir un identificador correcte o no tenir valor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prepara, dins la definició, un cursor que relaciona les amonestacions amb els alumnes, professors i tipus amonestacions amb una relació natural.
  - Prepara, dins la definició, un cursor que relaciona les amonestacions amb els alumnes, professors i tipus amonestacions amb una relació natural d'un alumne.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprova si el paràmetre d'entrada reportat té valor, si ho té, imprimeix les dades que obté el cursor preparat inicialment per obtenir les dades d'un alumne. Si no ho té, les imprimeix de les dades obtingudes al cursor preparat inicialment per obtenir les dades sense criteri de selecció.
  - El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.3.6. Llistat de totes les amonestacions**

Procediment que llista totes les amonestacions assignades.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_totes_amonestacions  
( paràmetre sortida <resultat>)
```

- Paràmetres d'entrada  
  
No té paràmetres d'entrada.
- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.



- Inicialitza la variable amb els paràmetres d'entrada per enregistrar el log.
- Executa el procediment de llistar amonestacions d'un alumne passant sense valor el paràmetre d'aquest. El procediment executat és *p\_llista\_amonestacio\_alumne*.
- Si el procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.3.7. Llistat de sancions d'un alumne**

Procediment que llista les sancions d'un alumne.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_llistat_sancio_alumne
( paràmetre entrada <idAlumne>,
  paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne.  
Pot tenir un identificador correcte o no tenir valor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prepara, dins la definició, un cursor que relaciona les sancions amb els alumnes i tipus de sanció amb una relació natural.
  - Prepara, dins la definició, un cursor que relaciona les sancions amb els alumnes i tipus de sanció amb una relació natural d'un alumne.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprova si el paràmetre d'entrada reportat té valor, si ho té, imprimeix les dades que obté el cursor preparat inicialment per obtenir les dades d'un alumne. Si no ho té, les imprimeix de les dades obtingudes al cursor preparat inicialment per obtenir les dades sense criteri de selecció.

- El procediment no ha tingut cap incidència insereix el log corresponent i retorna OK com a paràmetre de sortida.
- Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### 3.4.2.4. Mòdul estadístic

##### 3.4.2.4.1. Procediment p\_estadistica\_e1

Aquest procediment genera l'estadística que acumula el número d'amonestacions per alumne.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e1
( paràmetre entrada <idAlumne>,
  paràmetre entrada <inúmero>,
  paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne que rep l'amonestació.

<inúmero>  
El número que incrementa les amonestacions per alumne.  
Si és negatiu decrementa el valor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Comprova si l'alumne ja té un valor dins e1. Si no el té el crea amb el valor incremental, sempre que aquest sigui major que zero, si no ho és posa un zero.
  - Si l'alumne ja té valor el modifica incrementant o minvant segons el valor rebut.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### 3.4.2.4.2. Procediment p\_estadistica\_e2

Aquest procediment genera l'estadística que acumula el número de sancions per alumne i any.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e2
( paràmetre entrada <idAlumne>,
  paràmetre entrada <iexercici> ,
  paràmetre entrada <inúmero>,
  paràmetre sortida <resultat> )
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne que rep la sanció.

<iExercici>  
L'any en que s'ha produït la sanció.

<inúmero>  
El número que incrementa les sancions per alumne.  
Si és negatiu decrementa el valor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Comprova si l'alumne per aquest any ja té un valor dins e2. Si no el té el crea amb el valor incremental, sempre que aquest sigui major que zero, si no ho és posa un zero.
  - Si l'alumne ja té valor el modifica incrementant o minvant segons el valor rebut.
  - Com s'ha rebut una amonestació crida el procediment p\_estadistica\_e2\_e5, per comprova si el alumne que té més sancions ha canviat.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### 3.4.2.4.3. Procediment p\_estadistica\_e3

Aquest procediment genera l'estadística indica la mitjana d'amonestacions per professor i any.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e3  
( paràmetre entrada <idata>,  
  paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idata>  
La data de la amonestació.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prèviament és defineix el cursor que indicarà el número de amonestacions per professor en un exercici determinat.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Si l'any de la data de la sanció no té encara cap amonestació, s'inicialitza a zero.
  - Mitjançant el cursor predefinit, es calcula la mitjana per professor de l'any de l'amonestació.
  - Es modifica l'estadística amb el valor obtingut.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### 3.4.2.4.4. Procediment p\_estadistica\_e4

Aquest procediment genera l'estadística de sancions per curs i any.

- Execució del procediment

La sintaxi de l'execució del procediment és:

p\_estadistica\_e4  
( paràmetre entrada <idAlumne>,  
paràmetre entrada <idataSancio> ,  
paràmetre entrada <inúmero>,  
paràmetre sortida <resultat>)

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>  
L'identificador del alumne que rep la sanció.

<idataSancio>  
La data en que s'ha produït la sanció.

<inúmero>  
El número que incrementa les sancions per alumne.  
Si és negatiu decrementa el valor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada. Comprova el període a que correspon la data i el curs escolar matriculat l'alumne al període.
  - Comprova si el curs per aquest any ja té un valor dins e4. Si no el té el crea amb el valor incremental, sempre que aquest sigui major que zero, si no ho és posa un zero.
  - Si el curs a l'any ja té valor, el modifica incrementant o minvant segons el valor rebut.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.5. Procediment p\_estadistica\_e5**

Aquest procediment genera l'estadística del alumne més sancionat per any. Reb com a paràmetre l'alumne en qüestió.

- Execució del procediment

La sintaxi de l'execució del procediment és:

p\_estadistica\_e5

( paràmetre entrada <iExercici>,  
paràmetre entrada <idAlumne> ,  
paràmetre entrada <iValor>,  
paràmetre sortida <resultat>)

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<iExercici>

L'any en que s'ha produït la sanció.

<idAlumne>

L'identificador del alumne que rep la sanció.

<iValor>

El valor de sancions del alumne per aquest any.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Composa el nom del alumne com a "primer\_cognom segon\_cognom, nom".
  - Elimina els valors inferior al passat.
  - Comprova si l'alumne per aquest any ja té un valor dins e5. Si no el té el crea amb el valor i si el té el substitueix per el nou valor.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.6. Procediment p\_estadistica\_e2\_e5**

Aquest procediment, llançat dins el procediment p\_estadistica\_e2, comprova si la darrera sanció feta pot fer canvia el valor de l'estadística e5, i si així és crida a p\_estadistica\_e5 per actualitzar el valor.

- Execució del procediment

La sintaxi de l'execució del procediment és:

p\_estadistica\_e2\_e5

( paràmetre entrada <idAlumne> ,

paràmetre entrada <iExercici>,  
paràmetre sortida <resultat>)

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>

L'identificador del alumne que rep la sanció.

<iExercici>

L'any en que s'ha produït la sanció.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Comprova el número de sancions del alumne per aquest any.
  - Comprova el valor del alumne més sancionat del any.
  - Si el valor del alumne sancionat és igual o major al del alumne o alumnes més sancionats crida al procediment p\_estadistica\_e5 per la seva actualització.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.7. Procediment p\_estadistica\_e6**

Aquest procediment comprova del curs del que estat matriculat el darrer alumne sancionat, si el professor amb més sancions ha canviat. Si es així, crida al procediment p\_estadistica\_e6\_add per la seva actualització.

- Execució del procediment

La sintaxi de l'execució del procediment és:

p\_estadistica\_e6

( paràmetre entrada <idAlumne> ,  
paràmetre entrada <idata>,  
paràmetre sortida <resultat>)

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>

L'identificador del alumne que rep l'amonestació.

<idata>

La data en que s'ha produït l'amonestació.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prèviament es predefineix un cursor per obtenir el professors amb el seu número d'amonestacions fetes a alumnes matriculats a un curs escolar determinat.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada. Comprova el període a que correspon la data i el curs escolar matriculat l'alumne al període.
  - Mitjançant el cursor predefinit, actualitza el valor de e6 amb el dels professors que han fet més amonestacions, cridant al procediment p\_estadistica\_e6\_add.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.8. Procediment p\_estadistica\_e6\_add**

Aquest procediment genera l'estadística del professor que ha amonestat més per curs.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e6_add  
( paràmetre entrada <idCurs>,  
  paràmetre entrada <idPeriode> ,  
  paràmetre entrada <idProfessor> ,  
  paràmetre entrada <iValor> ,  
  paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<iCurs> i <idPeriode>



Els valors identificatiu d'un curs escolar.

<idProfessor>

L'identificador del professor que més ha amonestat.

<iValor>

El valor de amonestacions que ha fet el professor.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada.
  - Composa el nom del professor com a "primer\_cognom segon\_cognom, nom".
  - Elimina els valors inferior al passat.
  - Comprova si el professor per aquest curs escolar ja té un valor dins e6. Si no el té el crea amb el valor i si el té el substitueix per el nou valor.
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.9. Procediment p\_estadistica\_e7**

Aquest procediment genera l'estadística de la mitjana de sancions per alumne i curs escolar. Només s'actualitza el valor del curs escolar de la darrera sanció.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e7  
( paràmetre entrada <idAlumne>,  
  paràmetre entrada <idataSancio> ,  
  paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

Els paràmetres d'entrada són :

<idAlumne>

L'identificador del alumne que rep la sanció.

<idataSancio>

La data en que s'ha produït la sanció.

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Prèviament es defineix el cursor per obtenir el número de sancions de cada alumne matriculat a un curs escolar concret.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprovació paràmetres d'entrada. Comprova el període a que correspon la data i el curs escolar matriculat l'alumne al període.
  - Comprova si el curs escolar té un valor dins e7. Si no el té el crea amb una mitjana igual a zero.
  - Mitjançant el cursor predefinit, es calcula la mitjana de sancions per alumne i curs escolar.
  - Es modifica l'estadística amb el valor obtingut
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

#### **3.4.2.4.10. Procediment p\_estadistica\_e8**

Aquest procediment genera l'estadística amb el valor dels alumnes sense cap amonestació.

- Execució del procediment

La sintaxi de l'execució del procediment és:

```
p_estadistica_e8  
(paràmetre sortida <resultat>)
```

- Paràmetres d'entrada

No té paràmetres d'entrada

- Paràmetre de sortida <Resultat>

El paràmetre de sortida indica el resultat del procediment.

Si el procediment ha estat executat sense cap incidència el resultat del paràmetre de sortida serà OK, en cas contrari reportarà la incidència amb el format ERROR: <incidència> (Veure apartat **3.4.3 Errors reportats pels procediments.**)..

- Anàlisi funcional del procediment.
  - Inicialitza paràmetres de sortida i la variable amb els paràmetres d'entrada per enregistrar el log.
  - Comprova si hi ha qualche valor per aquesta estadística, si no n'hi ha el crea a zero.
  - Calcula el valor dels alumnes sense cap amonestació, restant al total d'alumnes el valor dels alumnes amonestats.
  - Es modifica l'estadística amb el valor obtingut
  - Finalment, hi ha el bloc que reporta les excepcions controlades dins el procediment.

### 3.4.3. Errors reportats pels procediments.

Missatge Reportat	Procediment generador	Descripció
<b>OK</b>	Tots	El procediment ha finalitzat correctament.
<b>ERROR: OPERACIO NO REALITZADA</b>	Tots	El procediment s'ha interromput per un error descongut.
<b>ERROR: Falta dada &lt;???</b>	Tots	El paràmetre d'entrada indicat a <???, no té valor.
<b>ERROR: &lt;???</b> no existeix	Tots	El paràmetre d'entrada <???) és un identificador que no existeix dins la seva taula corresponent.
<b>ERROR: &lt;???) ja existeix</b>	Tots	El registre insertat o modificat intenta assignar un valor al camp <???) que ja existeix a la taula.
<b>ERROR: Tipus document identitat incorrecte</b>	p_alumne_add p_alumne_update p_professor_add p_professor_update	El tipus de document identificatiu passat com a paràmetre d'entrada no esta dins la llista admesa: DNI, NIE, OTHER.

### 3.4.4. Implementació de procediments

#### 3.4.4.1. Manteniment de taules de persistència

Les taules de persistència es divideixen en 5 grans grups :

- Persones  
Permet l'enregistrament dels alumnes i professors.
- Cursos  
Manté l'estructura del pla d'estudis.
- Calendari  
Configura els horaris, matriculació i aules dins un curs escolar.
- Sancions  
Control d'amonestacions i sancions
- Avaluació  
Realitzar l'avaluació escolar dels alumnes.

A continuació, per cada grup, es detallen les taules i els seus procediments.

▪ **Grup Persones**

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Alumne	p_alumne_add	Per insertar un registre a la taula
	p_alumne_update	Per modificar un registre de la taula
	p_alumne_delete	Per eliminar un registre de la taula
Professor	p_professor_add	Per insertar un registre a la taula
	p_professor_update	Per modificar un registre de la taula
	p_professor_delete	Per eliminar un registre de la taula

**Taules auxiliars al Grup Persones**

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Pais	p_pais_add	Per insertar un registre a la taula
	p_pais_update	Per modificar un registre de la taula
	p_pais_delete	Per eliminar un registre de la taula

▪ **Grup Cursos**

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Titulacio	p_titulacio_add	Per insertar un registre a la taula
	p_titulacio_update	Per modificar un registre de la taula
	p_titulacio_delete	Per eliminar un registre de la taula
Curs	p_curs_add	Per insertar un registre a la taula
	p_curs_update	Per modificar un registre de la taula
	p_curs_delete	Per eliminar un registre de la taula
Assignatura	p_assignatura_add	Per insertar un registre a la taula
	p_assignatura_update	Per modificar un registre de la taula
	p_assignatura_delete	Per eliminar un registre de la taula

▪ **Grup Calendari**

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Horari	p_horari_add	Per insertar un registre a la taula
	p_horari_update	Per modificar un registre de la taula
	p_horari_delete	Per eliminar un registre de la taula
Festa	p_festa_add	Per insertar un registre a la taula
	p_festa_update	Per modificar un registre de la taula
	p_festa_delete	Per eliminar un registre de la taula
Aula	p_aula_add	Per insertar un registre a la taula
	p_aula_update	Per modificar un registre de la taula
	p_aula_delete	Per eliminar un registre de la taula
CursEscolar	p_cursEscolar_add	Per insertar un registre a la taula
	p_cursEscolar_delete	Per eliminar un registre de la taula
Matricula	p_matricula_add	Per insertar un registre a la taula
	p_matricula_update	Per modificar un registre de la taula
	p_matricula_delete	Per eliminar un registre de la taula
Classe	p_classe_add	Per insertar un registre a la taula
	p_classe_update	Per modificar un registre de la taula
	p_classe_delete	Per eliminar un registre de la taula

### Taules auxiliars al Grup Calendari

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Periode	p_periode_add	Per insertar un registre a la taula
	p_periode_update	Per modificar un registre de la taula
	p_periode_delete	Per eliminar un registre de la taula
Semanalectiva	p_semanaLectiva_add	Per insertar un registre a la taula
	p_semanaLectiva_update	Per modificar un registre de la taula
	p_semanaLectiva_delete	Per eliminar un registre de la taula
HorariDia	p_horariDia_add	Per insertar un registre a la taula
	p_horariDia_update	Per modificar un registre de la taula
	p_horariDia_delete	Per eliminar un registre de la taula
HorariSemana	p_horariSemana_add	Per insertar un registre a la taula
	p_horariSemana_update	Per modificar un registre de la taula
	p_horariSemana_delete	Per eliminar un registre de la taula
Sala	p_sala_add	Per insertar un registre a la taula
	p_sala_update	Per modificar un registre de la taula
	p_sala_delete	Per eliminar un registre de la taula

#### ▪ Grup Sancions

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
tipusSancio	p_tipusSancio_add	Per insertar un registre a la taula
	p_tipusSancio_update	Per modificar un registre
	p_tipusSancio_delete	Per eliminar un registre
tipusAmonestacio	p_tipusAmonestacio_add	Per insertar un registre a la taula
	p_tipusAmonestacio_update	Per modificar un registre
	p_tipusAmonestacio_delete	Per eliminar un registre
Sancio	p_sancio_add	Per insertar un registre a la taula
	p_sancio_update	Per modificar un registre
	p_sancio_delete	Per eliminar un registre
Amonestacio	p_amonestacio_add	Per insertar un registre a la taula
	p_amonestacio_update	Per modificar un registre
	p_amonestacio_delete	Per eliminar un registre

#### ▪ Grup Avaluació

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Avaluacio	p_avaluacio_add	Per insertar un registre a la taula
	p_avaluacio_update	Per modificar un registre de la taula
	p_avaluacio_delete	Per eliminar un registre de la taula

#### 3.4.4.2. Control de sancions

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Amonestacio i Sancio	p_amonestacio_sancio	Assigna sancio a una amonestació.
	p_amonestacio_control	Controla accions damunt regles per sancions.

### 3.4.4.3. Llistat dades bàsiques

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
Alumne	p_llistat_alumne	Llistat de les dades bàsiques d'un alumne.
	P_llistat_alumnes_curs	Llistat de les dades bàsiques dels alumns d'un curs corresponent a un període concret.
TipusAmonestacio	p_llistat_tipusA	Llistat de tots els tipus d'amonestacions
TipusSancio	p_llistat_tipusSancio	Llistat de tots els tipus de sanció
Sancio	p_llistat_sancio_alumne	Llistat de sancions d'un alumne concret.
Amonestacio	p_llistat_amonestacions	Llistat Amonestacions segons els paràmetres d'entrada de alumne o tipus d'amonestacions.
	p_llistat_totes_amonestacions	Llistat totes les amonestacions
	p_llistat_amonestacio_alumne	Llistat de amonestacions d'un alumne.

### 3.4.4.4. Mòdul estadístic

<i>Taula</i>	<i>Procediment</i>	<i>Descripció</i>
E1	p_estadistica_e1	Actualitza la taula E1
E2	p_estadistica_e2	Actualitza la taula E2
E3	p_estadistica_e3	Actualitza la taula E3
E4	p_estadistica_e4	Actualitza la taula E4
E5	p_estadistica_e5	Actualitza la taula E5
	p_estadistica_e2_e5	Comprova si debut a la darrera sanció, hi ha un nou alumne més sancionat. Cridat del procediment p_estadistica_E2
E6	p_estadistica_e6_add	Actualitza la taula E6
	p_estadistica_e6	Comprova si debut a la darrera amonestació, el curs en concret ha provocat que canviï el professor més amonestador.
E7	p_estadistica_e7	Actualitza la taula E7
E8	p_estadistica_e8	Actualitza la taula E8

#### 4. Producte final i jocs de proves

El producte final ha quedat format per la creació de les taules, seqüenciadors per l'assignació de identificadors a les taules que ho requereixen, i disparadors per actualitzar els identificadors segons els seqüenciadors.

Cada taula disposa dels procediments necessaris per gestionar el manteniment de les taules, per altes i baixes, i per modificació en cas de que tingui dades susceptibles de canviar.

El producte conté els procediments necessaris per mantenir les consultes requerides al plec tècnic del concurs.

Finalment, per gestionar les estadístiques demanades al plec, conté la creació de les taules i els procediments per mantenir-les.

Per la instal·lació del producte final, s'adjunta una guia d'instal·lació anomenada **Annex1. Instal·lació** amb els scripts de creació.

Per testejar el producte s'han creat uns jocs de proves que comproven la correcta execució de tots els procediments, tant els que mantenen els manteniments de les taules, per altes, modificacions i baixes, com els que realitzen les consultes o generen les estadístiques.

Aquests jocs de proves s'adjunten com annex al document **Annex 2. Proves** amb una explicació detallada per poder-los reproduir, juntament amb els fitxers que contenen les proves realitzades.

Les proves estan separades en 6 blocs:

- Configuració genèrica del centre. Creació pla d'estudis, cursos i assignatures.
- Configuració anual. Creació formació del calendari escolar, matriculacions, professors i tutories, formació d'aules.
- Sancions. Configuració d'amonestacions i sancions amb proves d'amonestacions i sancions, automàtiques i manuals. Aquest joc de proves executa la generació de les dades estadístiques a mida que es generen amonestacions i sancions als alumnes.
- Consulta dades estadístiques.
- Consultes. Execució dels procediments que retornen les consultes predefinides dins aquests.
- Execució de procediments amb dades incorrectes que provoquin la interrupció d'aquests reportant errors. Cada procediment comprova la no repetició de la inserció de dades o que la modificació la pugui motiva. Controla les dades obligatòries d'inserció i l'existència d'aquestes si pertanyen a altres taules. A més controla casos excepcionals com poden ser que les dates siguin incorrectes (la final menor que la inicial).

Pel tres primers blocs i el darrer, s'adjunten la relació de *logs* que generen les proves, a on es pot comprovar, l'execució correcta pel tres primers blocs i incorrecte amb el motiu pel darrer.

Per la visualització de dades estadístiques i realització de les consultes s'adjunta un fitxer *txt* amb les respostes obtingudes.

## 5. Cost del projecte

La valoració del projecte es realitza a continuació basant-se en criteris de costos acadèmics i no en preus de mercat.

Es parteix de la premissa que els programaris utilitzats son lliures o en llicència ja adquirida sense amortització pendent.

El projecte es realitza, segons la planificació en 80 dies, i com ja s'ha indicat un dia correspon a 2 h de treball, per tant,  $80 \times 2$  donen un total de 160 hores.

A la UOC un crèdit correspon a 15 hores lectives, això implica que la planificació del treball dona  $160 / 15 = 10,67$  crèdits.

El crèdit actualment té un cost de 15,73 €, per tant, 10,67 crèdits per 15,73 € dona un total de 167.84 €.

**El cost final, en criteris acadèmics del projecte és de 167,84 €.**

## 6. Conclusions

Dins el present projecte s'ha desenvolupat un disseny i posterior projecte d'implementació d'un sistema de gestió d'amonestacions i sancions per els centres educatius, el qual, no està preparat per una gestió conjunta si no individualitzada per cada centre.

El producte final desenvolupat, compleix les característiques tècniques del plec presentat per la Generalitat de Catalunya i és una opció vàlida per optar a l'atorgament del desenvolupament del sistema.

Fora de l'abast del plec de condicions, hi ha el desenvolupament de la gestió acadèmica dels alumnes, gestió que inicialment es contempla dins el disseny d'aquest producte. Seria molt recomanable el seu desenvolupament conjunt per una qualitat òptima del producte final.

La planificació del projecte, ha tingut com incidència el retràs de 7 dies de la pac 3, encara que aquest, no ha motivat el incompliment de l'entrega definitiva del producte final.

Per tant, com a conclusió que la planificació per la pac 2 i pac 3, encara que ben distribuïda, no tenia una distribució correcte de càrrega de treball, punt que ja es va avançar dins la avaluació dels riscos a la planificació.

## 7. Glossari

Bolonya. Procés iniciat al 1999 a la Declaració de Bolonya, per els ministres de educació de diversos països d'Europa, al qual es reforma els criteris europeus universitaris cap a la unificació i homologació d'aquests.

E/R. Diagrama de Entitat/Relació utilitzat per descriure gràficament els conjunts de entitats dels sistemes d'informació amb els seus atributs i les seves relacions.



ERP. Sistemes computacionals, o aplicacions funcionals, utilitzats per la planificació de recursos empresarials (*Enterprise resource planning*).

ETIG. Enginyeria Tècnica de Gestió

Forma Normal. Nivells per catalogar el grau d'inconsistència de les base de dades i de possibles anomalies lògiques.

Forma Normal de Boyce-Codd o FNBC. Compliment de la forma normal 3FN i els seus determinats són claus candidates.

Pàgina Web Dinàmica. Pàgines webs a on els seus continguts es generen al moment de la seva visualització.

Ple Tècnic. Document on es descriuen les condicions, necessitats i requeriments de caràcter obligatori per realitzar un projecte.

Script. Fitxer de ordres de comandes, en aquests cas de comandes de sql.

Sql. Structured Query Language o llenguatge de consulta estructura.

## 8. Bibliografia

Pla d'estudis de l'Enginyeria Tècnica de Gestió de la UOC.

Enunciat de TFC Base de dades relacionals segon semestre 2010-11.

*Mòdul 6. Teoria Normalització* de l'assignatura Base de Dades I.

Wikipedia. Enciclopedia de contingut lliure *on-line* en internet.

<http://en.wikipedia.org>    <http://es.wikipedia.org>

## 9. Annexos

Annex 1. Instal·lació

Document que especifica com instal·lar el producte.

Annex 2. Proves

Document que especifica els tests i proves realitzades i com repetir-les.

Carpeta: scripts

Contingut de fitxers sql amb la creació de les taules, seqüencials, disparadors i procediments.

Carpeta: proves

Contingut del joc de proves i tests realitzats segons les indicacions fetes al document Annex 2. Proves.