

**Disseny i implementació
d'un sistema de control d'un consultori mèdic.**

Memòria

Josep M^a Moragas Busquets

Gener 2006

Consultor: Alexandre Cornet

Índex

1. Anàlisi de l'enunciat	5
1.1. Estudi detallat de l'enunciat del projecte	5
1.1.1. Inici, com treballa un consultori?	5
1.1.2. Àrea d'aplicació	6
1.1.3. Altres	6
1.2. Identificació dels diferents grups de dades	6
1.2.1. Pacient	6
1.2.2. Metge	6
1.2.3. ATS	7
1.2.4. Administratiu	7
1.2.5. Hora de visita	7
1.2.6. Medicament	7
1.2.7. Fitxa de visita	7
1.2.8. Historial mèdic	7
2. Disseny de la base de dades	8
2.1. Disseny conceptual	8
2.1.1. Introducció	8
2.1.2. Sistema de treball	9
2.1.3. Accés al consultori mèdic	9
2.1.3.1. Accés a hores convingudes	9
2.1.3.2. Accés per urgències	10
2.1.4. Transcurs de la visita	10
2.1.4.1. Visita a hores convingudes	10
2.1.4.2. Urgències	10
2.2. Parts implicades	11
2.2.1. Actors i rols	11
2.2.1.1. Pacient	11
2.2.1.2. Metge / Metgessa	11
2.2.1.3. ATS (Assistent/a Tècnic Sanitari)	11
2.2.1.4. Administratiu	11
2.3. Documentació	12
2.3.1. Hora de visita	12
2.3.2. Fitxa de visita	12
2.3.2.1. Hora previst d'inici de visita	12
2.3.2.2. Identificació del pacient, del metge i ATS	12
2.3.2.3. Tipus de visita (concertada / urgència)	13
2.3.2.4. Observacions prèvies	13
2.3.2.5. Hora real d'inici de visita	13
2.3.2.6. Diagnòstic	13
2.3.2.7. Observacions	13
2.3.2.8. Llista de medicaments receptats	13
2.3.2.9. Historial mèdic	13
2.4. Entitat relació	13
2.4.1. Diagrama	14
2.4.2. Entitat i atributs	15
2.4.3. Aspectes del model E/R	16
2.5. Model Lògic	16
2.5.1. Regles	16
2.5.2. Transformació al model relacional	17
2.5.3. Observacions	18
3. Programació de la base de dades (Model Físic)	19
3.1. Creació dels Scripts (Procediments)	19
3.1.1. Programari utilitzat	19
3.1.2. Sentències SQL de creació de les taules	19
3.1.2.1. Taula PERSONA	20

3.1.2.2. Taula MEDICAMENT	20
3.1.2.3. Taula ESPECIALITAT	20
3.1.2.4. Taula ESTAT	21
3.1.2.5. Taula ACCES	21
3.1.2.6. Taula TORN	21
3.1.2.7. Taula DIASETMANA	21
3.1.2.8. Taula RECEPTA	22
3.1.2.9. Taula PACIENT	22
3.1.2.10. Taula METGE	22
3.1.2.11. Taula ATS	23
3.1.2.12. Taula ADMINISTRATIU	23
3.1.2.13. Taula HORAVISITA	23
3.1.2.14. Taula HORARI LABORAL	24
3.1.2.15. Taula FITXAVISITA	24
3.1.2.16. Taula PARAMETRE	24
3.1.2.17. Taula REGISTRELOG	25
3.1.2.18. Seqüències SQL per omplir taules	25
3.1.2.18.1. Dades per ESTAT	25
3.1.2.18.2. Dades per ACCES	25
3.1.2.18.3. Dades per TORN	25
3.1.2.18.4. Dades per DIASETMANA	25
3.1.2.18.5. Dades per PARAMETRE	26
3.1.2.18.6. Dades per METGR i ATS	26
3.1.3. Sentències SQL de definició de funcions	26
3.1.3.1. NOHIES	26
3.1.3.2. TEMPSVISITACORRECTE	26
3.1.3.3. METGEDISPONIBLE	26
3.1.3.4. ELIMINARRELACIONS	27
3.1.4. Sentències SQL de definició de procediments	27
3.1.4.1. REGISTRARLOG	27
3.1.4.2. CREAMPERSONA	27
3.1.4.3. MODIFICARPERSONA	27
3.1.4.4. ELIMINARPERSONA	28
3.1.4.5. CREAMPACIENT	28
3.1.4.6. MODIFICARPACIENT	28
3.1.4.7. CREAMMETGE	29
3.1.4.8. MODIFICARMETGE	29
3.1.4.9. CREAM ATS	29
3.1.4.10. MODIFICARATS	30
3.1.4.11. CREAMADMINISTRATIU	30
3.1.4.12. MODIFICARADMINISTRATIU	30
3.1.4.13. CREAMMEDICAMENT	31
3.1.4.14. MODIFICARMEDICAMENT	31
3.1.4.15. ELIMINARMEDICAMENT	31
3.1.4.16. CREAMESPECIALITAT	31
3.1.4.17. MODIFICARESPECIALITAT	32
3.1.4.18. ELIMINARESPECIALITAT	32
3.1.4.19. CREAMACCES	32
3.1.4.20. MODIFICARACCES	33
3.1.4.21. ELIMINARACCES	33
3.1.4.22. CREAMTORN	33
3.1.4.23. MODIFICARTORN	33
3.1.4.24. ELIMINARTORN	34
3.1.4.25. CREAMESTAT	34
3.1.4.26. MODIFICARESTAT	34
3.1.4.27. ELIMINARESTAT	35
3.1.4.28. INSERIRMEDICAMENTRECEPTA	35
3.1.4.29. ELIMINARMEDICAMENTRECEPTA	35
3.1.4.30. MODIFICARHORARILABORAL	35
3.1.4.31. ELIMINARHORARILABORAL	36

3.1.4.32.	CREARGUARDIA	36
3.1.4.33.	ELIMINARGUARDIA	37
3.1.4.34.	CREARHORAVISITA	37
3.1.4.35.	MODIFICARHORAVISITA	37
3.1.4.36.	ELIMINARHORAVISTA	38
3.1.4.37.	CREARFITXAVISITA	38
3.1.4.38.	MODIFICARFITXAVISITA	39
3.1.4.39.	ELIMINARFITXAVISITA	39
4.	Proves a la base de dades	39
4.1.	Inserció de dades	40
4.1.1.	PERSONA	40
4.1.2.	METGE	40
4.1.3.	ATS	40
4.1.4.	PACIENT	41
4.1.5.	HORARILABORAL	41
4.2.	CREARHORAVISITA	41
4.3.	MODIFICARHORAVISITA	42
4.4.	ELIMINARPERSONA	43
5.	Grandària de la base de dades	43
6.	Per acabar...	46
6.1.	Comentaris de disseny	46
6.2.	Proves	47
6.3.	Programari	47
6.4.	Fonts d'informació	47

1. Anàlisi de l'enunciat

Lectura i comprensió detallada de l'enunciat proporcionat.

1.1. Estudi detallat de l'enunciat del projecte

Veurem, primer de tot com treballa un Centre mèdic i l'abast que tindrà el nostre estudi.

1.1.1. Inici, com treballa un consultori?

Durant la lectura de l'enunciat, vas fent un esquema mental de com pot ser el funcionament d'un consultori mèdic, o d'un CAP (Centre d'assistència primària), que potser és el que ens queda més a prop i que tots mes o menys hem passat per ell. Intentem seguir els passos que es duen a terme durant una visita, des que entrem fins a l final, relacionant els diferents apartats de l'enunciat amb fets reals.

Un cop acabada la lectura segueix el treball imaginatiu, ara molt més detallat. Perquè hi vaig anar l'última vegada?, si, la revinclada al turmell, ja fa bastant temps però no crec que el sistema hagi variat gaire. Farem una descripció dels fets:

- **Accés:** va ser concertant visita, ja que la revinclada acabada de fer (en calent) no fa molt de mal, és passat unes hores quan el mal es fa present de debò. Recordo que a primera hora del matí vaig trucar demanant hora pel metge de capçalera que tenia assignat, el personal administratiu em va donar un número de visita i una hora aproximada de realització, que per cert era bastant tardana. Ara ja estava a punt per anar al CAP.
Cal recordar que hi ha tres camins d'accés al consultori, o en aquest cas al CAP:
 - Previ obtenció de número de visita, que variarà depenent de la quantitat de malalts que hi hagi. Això condicionarà l'hora de visita que tindrem concertada, també influirà la puntualitat dels pacients, ja que un retràs o absència d'un d'ells suposarà avançar la tanda i sobre tot la dedicació del metge a cada visita.
 - Hora concertada, depenent del CAP s'ofereixen serveis que requereixen una sol·licitud d'hora avançada, com per exemple una visita al ginecòleg per la revisió periòdica, posem pel cas. Això vol dir que si la revisió és semestral, les visites es concertin cada sis mesos, una vegada acabada ja es fixi la data per la pròxima.
 - Per urgències, en cas d'un contratemps o malaltia greu que necessiti d'atenció immediata, ens presentem al CAP sense avís previ i serem atesos el més aviat possible pel metge que estigui de guàrdia en aquell moment, sigui o no el metge de capçalera que nosaltres tenim assignat.
- **Espera:** quan arribem al CAP, ens dirigim al personal administratiu, que ens recull les dades personals, prepara el nostre expedient, ens indica el número de visita que s'està realitzant actualment i ens mostra la sala d'espera.
- **Visita:** quan ens toca el torn, normalment molt més tard que el concertat, exposem el problema al metge, que realitza el diagnòstic i si la solució està al seu abast ens recepta els medicaments oportuns, si no ens remet a un altre centre amb els medis necessaris. Tots aquests passos queden registrats a la fitxa de visita, que passa a formar part del nostre historial mèdic. Tornant al cas pràctic, el diagnòstic va ser un embenat de compressió al turmell i la visita ràpida al hospital.
- **Sortida:** ens apropem de nou al taulell del personal administratiu, per indicar el final de la visita i si es dona el cas, la realització del document 'recepta' de tots els medicaments que ens ha receptat el metge i ofereixen descompte, per part de la Seguretat social, en el moment de compra de a la farmàcia.

Aquests son, presentats en esquema, els passos que seguim durant una visita al consultori mèdic i el punt de partida per afrontar el projecte segons l'enunciat.

1.1.2. Àrea d'aplicació

Per a la realització del projecte i com molt be diu l'enunciat: realitzar el disseny d'un sistema informàtic capaç de dur *part* de la gestió d'un consultori mèdic, ens centrarem en la gestió de la relació del pacient amb el consultori mèdic, partint del concepte d'historial mèdic, amb totes les fitxes de visita que el componen i de tots els processos que comporten una visita mèdica, des de l'establiment de l'hora de visita fins la sortida.

Altres punts que queden fora de l'abast del projecte, però que contribueixen al funcionament del centre i que poden ser tractats en futures revisions serien:

- Gestió del personal intern del centre, entre d'altres, la gestió d'incorporacions, acomiadaments, jubilacions, vacances, sous...
- Gestió del material sanitari usat al centre, consum de medicaments, desinfectants, venes, i tot el material divers que es pot utilitzar durant la visita, a més del seu inventari i compra.
- Gestió del material divers, consumibles d'oficina, electricitat, gas, dietes, mobiliari...
- Gestió del manteniment de les instal·lacions, que inclouria la relació amb empreses de manteniment com lampistes, paletes, fusters...

1.1.3. Altres

L'objectiu del projecte és la creació de les estructures de dades necessàries, així com els mètodes d'accés a aquestes dades. No contempla el disseny de la interfície d'usuari, ni el programari d'instal·lació del servidor ni client.

1.2. Identificació dels diferents grups de dades

Definit l'abast del projecte i havent descrit el funcionament d'un centre qualsevol per la part que ens toca, podem definir quins son els objectes que participaran en el projecte, que formen part del mon real i son necessaris per els nostres objectius:

1.2.1. Pacient

Malalt, persona que acudeix al centre per trobar solució als seus problemes, pot accedir al centre demanant hora de visita, o si el problema és de caire urgent, pot accedir per urgències. Cada pacient té assignat un metge de capçalera, que coneix tot el seu historial mèdic. En cas d'accedir per urgències pot ser atès per el metge de guàrdia.

1.2.2. Metge

Metge especialista:

Pertany al personal que treballa al consultori, la seva tasca consisteix a visitar als pacients, efectuar un diagnòstic i receptar els medicaments que cregui necessaris pel guariment del malalt. Ha de conèixer l'historial mèdic de cada persona que el visita. També disposar de la informació dels medicaments que té al seu abast per receptar al malalt. Després de cada visita registrarà en una fitxa el contingut del diagnòstic i els medicaments receptats. El metge especialista no fa guàrdies.

Metge de capçalera:

Pertany al personal que treballa al consultori, te assignats una quantitat determinada de pacients, la seva tasca consisteix a visitar als pacients, efectuar un diagnòstic i receptar els medicaments que cregui necessaris pel guariment del malalt. Ha de conèixer l'història mèdica de cada persona que te al seu càrrec. També disposar de la informació dels medicaments que té al seu abast per receptar al malalt. Després de cada visita registrarà en una fitxa el contingut del diagnòstic i els medicaments receptats. El metge de capçalera realitza guàrdies.

1.2.3. ATS

Pertany al personal que treballa al consultori, cada ATS està assignat a un metge, realitza funcions d'assistència durant les visites. Si està assignat a un metge de capçalera, també realitza guàrdies.

1.2.4. Administratiu

Pertany al personal que treballa al consultori, rep les peticions de visites mèdiques i les gestiona. Depenent del tipus de visita assigna dia, hora i metge al pacient i s'encarrega de que es dugui a terme quan toca.

1.2.5. Hora de visita

Establiment de la trobada entre el metge i el pacient, és creada pel personal administratiu en funció de les hores disponibles del metge, la durada de la qual vindrà en funció del mínim establert entre cada visita. L'espai màxim entre visites l'establirà el personal administratiu previ coneixement de cada motiu de visita.

1.2.6. Medicament

Composició normalment química, que recepta el metge i que te unes propietats que ajuden al guariment del malalt. Existeixen gran quantitat de medicaments en el mercat, però cada metge te els seus preferits que son els que recepta als malalts i que té al seu abast les indicacions, contradiccions i administració.

1.2.7. Fitxa de visita

Després de cada visita, el metge registra el contingut, la descripció del problema, el diagnòstic, observacions i els medicaments receptats. Aquí queda constància del pas del malalt pel consultori.

1.2.8. Història mèdica

Totes les vegades que el malalt ha necessitat dels serveis del consultori mèdic queden registrades, així és fàcil pel personal sanitari reconstruir tots els problemes que ha tingut el pacient. D'aquesta manera, amb l'estudi de precedents pot ajudar a la realització de futurs diagnòstics i a solucionar problemes de salut més ràpidament.

2. Disseny de la base de dades

Per al correcte disseny d'una base de dades relacional, cal tenir presents diversos criteris. Els tres primers fan referència a l'organització:

- Qualsevol aparellament de dades (una persona amb totes les dades personals...) ha de constar només una sola vegada en una sola taula de la base de dades. Així, quan l'entrem ho fem en un sol lloc i si mai hem de modificar-la també ho farem en un sol lloc.
- Qualsevol dada que es pugui deduir inequívocament de dues altres dades preexistents ha de ser exclosa de la base de dades. Els càlculs es fan en l'explotació de la base de dades, però no formen part de les dades contingudes.
- Com a conseqüència dels dos punts anteriors, les taules només poden contenir:
 - Dades no repetides ni deduïbles.
 - D'altres referències que lliguin aquestes dades amb les contingudes en altres taules.

També hi ha condicions lligades al programa informàtic, el motor de la base de dades:

- L'organització de les dades en taules ha de ser compatible amb les capacitats de processament del programa informàtic de què disposem.
- L'explotació de les dades ha de permetre la presentació del resultat de la combinació de les diverses taules: el caràcter relacional no pot limitar-se a una simple estructuració teòrica, sinó que ha de donar lloc a informacions presentades d'una manera pràctica.

2.1. Disseny conceptual

L'esquema conceptual és una descripció detallada dels requeriments en quant a informació dels usuaris, conté descripcions dels tipus de dades, restriccions i relacions entre elles.

Pel model conceptual hem escollit una modelització basada en un disseny E/R (Entitat/relació) que descriu les dades com a entitats, vincles (relacions) i atributs.

El model E/R va ser proposat per Peter P. Chen entre els anys 1976-1977.

Després, altres autors han investigat i escrit sobre el model, proporcionant importants aportacions. És per aquesta raó, que no és pot considerar que existeixi un únic model E/R.

El model E/R descriu les dades com a entitats, relacions (vincles) i atributs i permet representar l'esquema conceptual d'una base de dades de forma gràfica mitjançant els diagrames E/R.

2.1.1. Introducció

Definim com funcionarà el nostre consultori mèdic, analitzant cadascuna de les parts implicades, entitats i processos. Després de llegir l'enunciat i de fer l'anàlisi, tenim una lleugera idea del funcionament d'un consultori mèdic, al qual hem fet referència com si d'un CAP es tractés. El nostre mantindrà aquesta estructura però funcionarà de manera independent, serà un centre únic que no formarà part de cap sistema de consultes, cal dir que la base de dades resultant podria formar part d'un sistema distribuït, amb una base de dades única accessible per un software remot, només caldria identificar la informació en funció del centre (lloc) al qual pertany.

Considerarem un servidor de dades al mateix centre, amb una sèrie de clients que podran accedir-hi des de qualsevol lloc de les instal·lacions on faci falta.

Ara només cal concretar, després de llegir l'enunciat i d'analitzar el sistema de treball d'un CAP, com funcionarà el nostre consultori mèdic.

2.1.2. Sistema de treball:

El nostre centre mèdic dividirà la seva activitat en dos blocs, servei de consulta mèdica i servei d'urgències:

Servei de consulta mèdica:

Estarà format pel conjunt de metges amb els corresponents ATS (Assistent Tècnic Sanitari), cadascun d'ells especialitzat en un àrea en concret. També el personal administratiu. Aquest col·lectiu s'encarregarà d'atendre els pacients que demanin hora de visita amb anticipació.

Servei d'urgències:

Estarà format com a mínim d'un metge amb un ATS i un administratiu, que estaran capacitats per atendre els imprevistos que puguin sorgir en un moment donat. Si es creu convenient, es podrà afegir més personal mèdic per atendre una gran demanda en un període concret.

Descriurem totes les situacions que es poden donar en el nostre consultori mèdic, des de que el pacient hi accedeix fins que en surt.

2.1.3. Accés al consultori mèdic:

Tindrem en compte dos tipus d'accés al consultori, mitjançant hores convingudes i l'altra a través del servei d'urgències que ofereix el nostre consultori mèdic.

2.1.3.1. Accés a hores convingudes:

El pacient es dirigirà al personal administratiu del Consultori Mèdic, a partir d'ara CM, per obtenir una data i hora de visita amb el metge requerit segons l'origen del problema. El personal administratiu, segons la disponibilitat horària del metge donarà dia i hora de visita al pacient. Per tant el personal administratiu coneixerà els horaris de treball de tots els metges i també si el metge està disponible en el moment desitjat.

Metge de capçalera:

El pacient obtindrà hora amb el seu metge de capçalera quan, desconeixent l'origen del problema, requereix un primer diagnòstic. Les hores de visita seran concertades al mateix dia, s'aniran adjudicant les hores a mida que es vagin sol·licitant.

Metge especialista:

El pacient coneix l'origen de la seva patologia, per tant requereix un metge especialitzat que li solucioni el problema. Les hores de visita s'adjudicaran en funció de la disponibilitat del metge.

Cada pacient tindrà un metge de capçalera assignat, quan hi ha un problema el pacient ha de demanar hora de visita al metge de capçalera, el qual li farà el diagnòstic. Si la solució del problema no està al seu abast, suggerirà al pacient la visita a un metge especialista, el qual disposant d'uns coneixements més concrets en quan a la malaltia, podrà oferir ajut al pacient. Llavors el pacient es tornarà a dirigir al personal administratiu per obtenir una visita al metge especialista.

El personal administratiu distribuirà les hores de visita depenent de cada cas, hi ha un temps mínim de visita establert, després aquest temps s'ampliarà en funció del problema, aquest criteri es deixarà en mans del personal administratiu, que fent ús de la seva experiència o previ consulta al metge indicat.

D'aquesta manera obtindrem una cita per un dia determinat de trobada entre un pacient i un metge. També contindrà si cal, observacions que puguin ser útils pel metge en el moment de la visita.

2.1.3.2. Accés per urgències:

El CM ofereix un servei d'atenció d'urgències, això vol dir que com a mínim hi haurà una persona d'administració, un metge disponible i un ATS al CM per atendre qualsevol imprevist. Un accés per urgència no té hora convinguda, per tant el pacient es presenta al CM i el personal administratiu recull la informació necessària, que és transmesa al metge i atén al pacient tan aviat com és possible.

2.1.4. Transcurs de la visita

Podrem distingir dos tipus de visita, condicionats pel tipus d'accés del pacient al CM, el convencional amb hores convingudes i la urgència. Una vegada finalitzada la visita, s'obtindrà un registre de l'activitat de la visita en forma de Fitxa de visita, on hi haurà tots els detalls, inclòs el tipus d'accés al CM.

2.1.4.1. Visita a hores convingudes

El metge podrà consultar les hores de visita i pacients que té assignat en un període de temps concret. També tindrà al seu abast tot l'historial mèdic del pacient que desitgi.

Quan el pacient arribi a la consulta, el metge sabrà quin dia va ser concertada aquella visita i a quina hora estava prevista, també les observacions pertinents que el personal administratiu hi haurà inclòs.

A partir d'aquesta informació el metge encetarà una nova fitxa de visita referent a aquesta consulta, on hi haurà indicat:

- L'hora previst d'inici de la visita.
- El nom del pacient
- El nom del metge.
- El nom de l'ATS.
- Les observacions prèvies indicades pel personal administratiu

Ara el metge podrà indicar l'hora real d'inici de la visita.

Una vegada s'hagi realitzat la visita i el metge realitzi el diagnòstic, podrà incloure'l a la fitxa, amb tots els detalls que siguin necessaris que cregui convenient i d'informació rellevant. També podrà incloure-hi els medicaments receptats per fer front al problema. Al finalitzar la visita caldrà indicar l'hora i tancar la fitxa de visita.

2.1.4.2. Urgències

El metge podrà consultar, en el moment d'atendre el pacient d'urgència, les dades que moments abans haurà recollit el personal administratiu:

- El nom del pacient.
- El nom del metge.
- El nom de l'ATS.
- Les observacions prèvies indicades pel personal administratiu.

També podrà consultar l'historial mèdic del pacient.

A partir d'ara es procedirà igual que si la visita fos concertada, obtenint com a resultat final una fitxa de visita. Les diferència serà que no existirà una hora de visita prèvia, es crearà en el moment que arribi la urgència al CM.

2.2. Parts implicades

A mesura que anem avançant, podem concretar més ens els rols dels integrants, persones i objectes del nostre CM. Ara podem definir la seva tasca i informació necessària.

2.2.1. Actors i rols

Descriurem què és cada part i definirem el seu paper.

2.2.1.1. Pacient

Persona que estarà donada d'alta i podrà gaudir dels serveis del CM, tindrà un número d'identificació propi i a part de conèixer les seves dades personals, tindrà un metge de capçalera assignat. El CM tindrà tot el seu historial mèdic. Podrà accedir al servei mitjançant visita concertada i directament si presenta una urgència.

2.2.1.2. Metge / metgessa

Forma part del personal que treballa al CM, un metge pot estar especialitzat en un camp o varis, o pot estar especialitzat en medicina general i ser metge de capçalera. Cada metge treballa amb un ATS. Treballa a jornada intensiva, en torns de matí o tarda de sis hores de duració i que comencen a les nou hores o a les quinze hores respectivament, o guàrdia de tot un dia, una guàrdia dura vint i quatre hores i comença a les quinze hores. Quan un metge està de guàrdia, interromp el seu servei de consultes i es dedica a la guàrdia exclusivament. Independentment si treballa de torn de tarda o matí, estarà dos dies dedicat exclusivament a la guàrdia.

Un metge pot tenir o no pacients assignats, un especialista no tindrà pacients assignats, tots els pacients previ hora de visita poden accedir als seus serveis. Un metge de capçalera tindrà pacients assignats.

El CM coneix els horaris de treball de tots els metges integrants, també coneix si un metge està en actiu o presenta algun tipus de baixa, també si està de vacances o serà l'encarregat d'atendre les urgències durant un període, el qual podem dir que està de guàrdia.

2.2.1.3. ATS (Assistent/a Tècnic Sanitari)

Forma part del personal que treballa al CM, cadascun d'ells està assignat a un metge al qual donarà suport. Treballa a jornada intensiva, en torns de matí o tarda de sis hores de duració, o guàrdia de tot un dia, una guàrdia dura vint i quatre hores i comença a les quinze hores, al igual que el metge interromp la seva jornada laboral de treball per dedicar-se exclusivament a la guàrdia.

El CM coneix els horaris de treball de tots els ATS integrants, també coneix si un ATS està en actiu o presenta algun tipus de baixa, també si està de vacances o estarà al servei d'urgències durant un període, el qual podem dir que està de guàrdia.

2.2.1.4. Administratiu

Forma part del personal que treballa al CM, és l'encarregat de rebre als pacients. Treballa a jornada intensiva, en torns de matí o tarda, o guàrdia de tot un dia, amb els mateixos horaris que els metges. Segons la demanda del pacient li proposa un dia i hora de visita, que en cas de ser acceptada la formalitza al sistema. S'encarregarà d'actualitzar l'agenda de la setmana, indicant per cada treballador (metges / administratius), la seva activitat diària prevista, horari de visites o de treball i les guàrdies assignades. També coneixerem el seu estat: actiu, baixa, vacances... i horari de treball.

Si treballa atenent urgències, direm que està de guàrdia i s'encarregarà de gestionar-les perquè el metge les pugui atendre el més aviat possible.

2.3. Documentació

Aquesta activitat del CM generarà informació que serà recollida pels següents objectes, s'anomena documentació perquè en el món real pot ser presentada d'aquesta manera.

2.3.1. Hora de visita

Quan un pacient es dirigeix al CM per requerir els seus serveis, en funció de les seves necessitats, el personal Administratiu crea l'hora de visita que està composta de les següents dades:

- Identificació del pacient.
- Metge.
- ATS.
- Dia i hora de visita.
- Observacions.

El personal administratiu assigna un metge al pacient, que tots dos estaran identificats pel nom i número d'identificació, que en una data i hora en concret el podrà atendre i intentar solucionar els seus problemes. A més hi hauran, si és necessari unes observacions relatives a la demanda. El metge implicat podrà conèixer aquesta informació, així com tenir una relació de hores de visita durant un període de temps.

Quan es tracti d'una visita concertada, l'hora de visita es generarà un temps abans per organitzar l'agenda del metge, que el personal administratiu coneixerà.

En cas d'una urgència, el metge i personal administratiu que atindrà al pacient seran els que estiguin de guàrdia en aquell moment, l'administratiu crearà l'hora de visita en el moment de recepció del pacient.

2.3.2. Fitxa de visita

Recollirà tota la informació generada durant una visita, el punt de partida de creació de la fitxa de visita serà l'hora de visita. La col·lecció de fitxes de visita d'un pacient, representa el seu historial mèdic. S'identificarà amb un codi. A partir d'una hora donada de visita el metge crea una fitxa de visita, que estarà composta per:

- L'hora previst d'inici de la visita.
- LA identificació del pacient.
- La identificació del metge.
- La identificació del ATS.
- Tipus de visita (concertada / urgència)
- Les observacions prèvies indicades pel personal administratiu.
- L'hora real d'inici de la visita.
- El diagnòstic.
- Les observacions.
- La llista de medicaments receptats.

2.3.2.1. Hora previst d'inici de la visita

Serà l'hora que establirà l'administratiu quan el pacient li proposi una cita amb un metge determinat, ja sigui de capçalera o un especialista. Aquesta dada provindrà de l'hora de visita.

2.3.2.2. Identificació del pacient, del metge i ATS

Només caldrà indicar el nom del pacient i el seu número d'identificació. Aquestes dades provindran de l'hora de visita.

2.3.2.3. Tipus de visita (concertada / urgència)

Aquí s'identificarà el tipus d'accés del pacient al CM, si ha estat per mitjà de visita concertada o ha accedit per urgències.

2.3.2.4. Observacions prèvies

Aquesta dada provindrà de l'hora de visita. El personal administratiu anotarà les observacions que cregui oportunes referents a la petició del malalt durant la creació de l'hora de visita. Aquestes dades poden ser importants pel metge i pel desenvolupament de la visita.

2.3.2.5. Hora real d'inici de la visita

Una cosa és la hora prevista i l'altra és la hora real en que comença la visita. La comparació d'aquestes dues dades poden servir per la creació d'estadístiques, per fixar l'horari mínim de visita per exemple entre d'altres.

2.3.2.6. Diagnòstic

Veredictes del metge, aquí s'anotarà tota la informació mèdica referent al problema presentat pel pacient, en cas de ser la visita realitzada pel metge de capçalera, si el problema ha de ser analitzat per un especialista, també hi haurà la informació referent al altre metge que ha de realitzar el diagnòstic. També s'hi indicarà el tractament que ha de rebre el pacient.

2.3.2.7. Observacions

Aquí hi haurà tota la informació referent a la visita que cal tenir en compte per a properes intervencions, també fets rellevants que pot ser usada per altres metges especialistes que poden tractar al pacient, o simplement fets que no podem deixar passar per alt.

2.3.2.8. Llista de medicaments receptats

Si el diagnòstic implica prendre medicaments, aquí hi haurà el llistat. El metge receptarà medicaments que ell coneix i creu oportuns per guarir al pacient. Les propietats i característiques d'aquests medicaments seran conegudes pels metges del CM. No es receptaran medicaments que no formin part del llistat de medicaments acceptats pel CM.

2.3.2.9. Historial mèdic

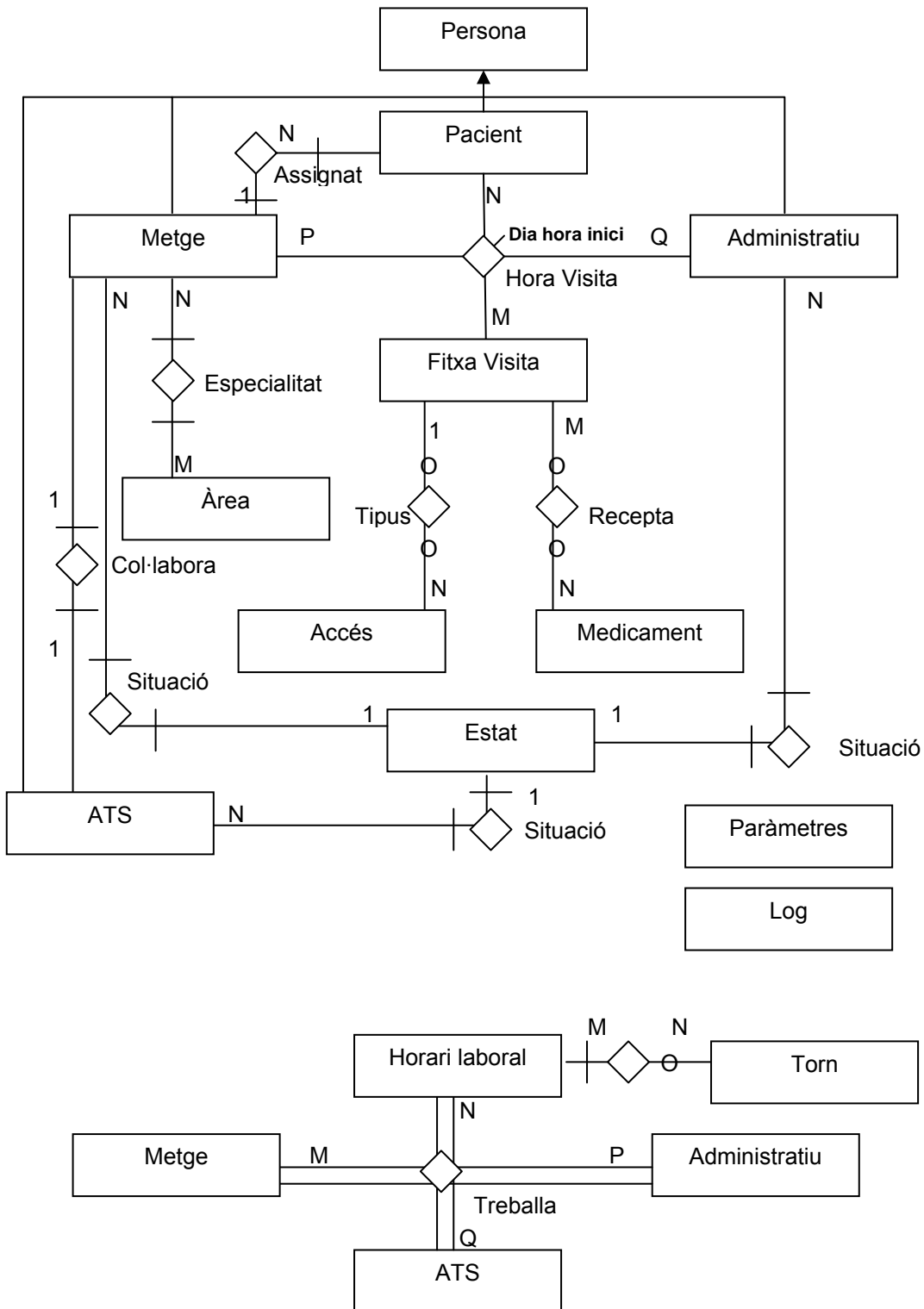
Cada pacient tindrà un historial mèdic, on hi haurà descrit totes les visites realitzades pel CM. L'historial estarà format per recull de totes les fixes de visita ordenades cronològicament. Aquest historial servirà al metge per conèixer l'evolució de l'estat de salut del pacient.

2.4. Entitat / Relació

Després de descriure el funcionament del nostre CM, fem la modelització de les dades i reflectim el model conceptual. Els elements essencials del model, són les entitats, els atributs y les relacions entre les entitats. Una entitat és un objecte que existeix y que és distingible d'altres objectes, per exemple el metge Joan. Les entitats tenen atributs. Un atribut d'una entitat és una característica interessant sobre ella, es a dir, representa alguna propietat que ens interessa guardar. Per exemple, en Joan té una data de naixement i un lloc de residència. Pot tenir també atributs que pot compartir amb altres persones.

2.4.1. Diagrama

La figura següent mostra un diagrama ER que satisfà els requisits definits anteriorment, descrits en funció de les entitats:



Cal remarcar la inclusió de les entitats Paràmetres i Log. Paràmetres permetrà definir totes aquelles dades preestablertes que seran utilitzades pel personal del CM en el transcurs de la seva feina. Log registrarà tots els moviments de la base de dades.

2.4.2. Entitats i atributs

Els atributs de les entitats que figuren al diagrama són els següents (les claus primàries s'han subratllat):

- **Persona**

Identificacio, nom, cognom1, cognom2, NIF, Carrer, numero, pis, porta, codi_postal, poblacio, telefon

- **Pacient** (Entitat subclasse de persona)

Identificacio

- **Metge** (Entitat subclasse de persona)

Identificacio

- **ATS** (Entitat subclasse de persona)

Identificacio

- **Administratiu** (Entitat subclasse de persona)

Identificacio

- **Fitxa_visita**

codi, diagnostic, observacions

- **Area**

Area

- **Accés**

tipus accés

- **Medicament**

referencia, nom, fabricant, familia, indicacions, contradiccions, administracio

- **Estat**

estat

- **Paràmetres**

nom, valor

- **Log**

codi, procés, data, hora, parametre_entrada, parametre_sortida, RSP

- **Horari_laboral** (entitat dèbil, s'identifica completament amb la identificació de la persona)

Dia, torn, guardia

2.4.3. Aspectes del model E / R

Hi ha una generalització / especialització de l'entitat *Persona*, les persones que hi ha al cap poden ser de quatre tipus, Pacient, que té un metge de capçalera assignat, metge amb especialitat, l'estat laboral en que es troba i un ATS que hi col·labora, ATS que col·labora amb un metge i també coneixem el seu estat laboral i finalment la persona administrativa de la qual coneixem el seu estat laboral.

La interrelació de quatre entitats *Hora de visita*, associa *metge*, *pacient*, *administratiu* i la *fitxa de vista*, és el punt de sortida de la visita d'un *pacient* al CM, ja que a partir d'aquesta cita s'associen les entitats per generar la *Fitxa de vista*, que és la culminació del treball.

Així mateix la *fitxa de visita* mostra el tipus d'accés del malalt al centre (visita o urgència) i també està associada a una llista de *medicaments* receptats pel metge.

Cal destacar l'entitat dèbil horari laboral que necessita estar associada a una persona (metge, ATS, administratiu). L'horari laboral està formada pels dies de la setmana, els torns que divideixen aquests dies i si existeix guàrdia, però no s'acaba de definir fins que no hi associem una persona que compleixi els torns i guàrdies descrits.

2.5. Model Lògic

Partirem del resultat de l'etapa del disseny conceptual expressat mitjançant el model ER i el transformarem en una estructura de dades del model relacional.

2.5.1. Regles

Els elements bàsics del model ER són les entitats i les interrelacions:

- Les entitats, quan es tradueixen al model relacional, originen relacions (anomenades taules a la majoria dels SGBD del mercat).
- Les interrelacions, en canvi, quan es transformen, poden donar lloc a claus foranes d'alguna relació ja obtinguda o poden donar lloc a una nova relació.

A continuació, presentem en forma de taula els aspectes més bàsics i que ens afecten en el nostre model E/R i la transformació que poden rebre:

Element del model ER	Transformació al model relacional
Entitat	Relació
Interrelació 1:1	Clau forana
Interrelació 1:N	Clau forana
Interrelació M:N	Relació
Interrelació n-ària	Relació
Interrelació recursiva	Com a les interrelacions no recursives: <ul style="list-style-type: none"> • Clau forana per a binàries 1:1 i 1:N • Relació per a binàries M:N i n-àries
Entitat dèbil	La clau forana de la interrelació identificadora forma part de la clau primària
Generalització/ especialització	<ul style="list-style-type: none"> • Relació per a l'entitat superclasse • Relació per a cadascuna de les entitats subclasse
Entitat associativa	La transformació de la interrelació que l'origina és alhora la seva transformació

2.5.2. Transformació al model relacional

Seguint les transformacions descrites en el quadre anterior:

- **PERSONA** (idPersona, nomPersona, cognom1Persona, cognom2Persona, edatPersona, nifPersona, carrerPersona, numeroPersona, pisPersona, portaPersona, cpPersona, poblacióPersona, telefonPersona)
- **MEDICAMENT** (refMedicament, nomMedicament, fabricantMedicament, familiaMedicament, indicacionsMedicament, contradiccionsMedicament, administracioMedicament)
- **ESPECIALITAT** (idEspecialitat, nomEspecialitat)
- **ESTAT** (idEstat, nomEstat)
- **ACCES** (idAcces, tipusAcces)
- **TORN** (idTorn, nomTorn)
- **DIASETMANA** (idDiasetmana, nomDiasetmana)
- **RECEPTA** (idRecepta, fitxavisitaRecepta, medicamentRecepta)
 - on { fitxavisitaRecepta } referencia FITXAVISITA (idFitxavisita)
 - on { medicamentRecepta } referencia MEDICAMENT (refMedicament)
- **PACIENT** (idPacient, metgeAssignat)
 - Subclasse de PERSONA
 - on { idPacient } referencia PERSONA (idPersona),
 - on { metgeAssignat } referencia METGE (idMetge)
- **METGE** (idMetge, especialitatMetge, estatMetge, atsMetge)
 - Subclasse de PERSONA
 - on { idMetge } referencia PERSONA (idPersona),
 - { especialitatMetge } referencia ESPECIALITAT (idEspecialitat),
 - { estatMetge } referencia ESTAT (idEstat),
 - { atsMetge } referencia ATS (idAts)
- **ATS** (idAts, estatAts, metgeAts)
 - Subclasse de PERSONA
 - on { idAts } referencia PERSONA (idPersona),
 - { estatAts } referencia ESTAT (idEstat),
 - { metgeAts } referencia METGE (idMetge)
- **ADMINISTRATIU** (idAdministratiu, estatAdministratiu)
 - Subclasse de PERSONA
 - on { idAdministratiu } referencia PERSONA (idPersona),
 - { estatAdministratiu } referencia ESTAT (idEstat)
- **HORAVISITA** (idHoravisita, datahorainiciHoravisita, datahorafiHoravisita, metgeHoravisita, pacientHoravisita, observacionsHoravisita)
 - on { metgeHoravisita } referencia METGE (idMetge),
 - { pacientHoraVisita } referencia PACIENT (idPacient)

- **HORARILABORAL** (idHorarilaboral, personaHorarilaboral, diaHorarilaboral, tornHorarilaboral, guardiaHorarilaboral)
 - on {personaHorarilaboral} referencia PERSONA (idPersona),
 - on {diaHorarilaboral} referencia DIASETMANA (idDiasetmana),
 - on {tornHorarilaboral} referencia TORN (idTorn)
- **FITXAVISITA** (idFitxavisita, pacientFitxavisita, metgeFitxavisita, atsFitxavisita, horavisitaFitxavisita, accesFitxavisita, horainiciFitxavisita, horafiFitxavisita, diagnosticFitxavisita, observacionsFitxavisita)
 - on {pacientFitxavisita} referencia PACIENT (idPacient),
 - on {metgeFitxavisita} referencia METGE (idMetge),
 - on {atsFitxavisita} referencia ATS (idAts),
 - on {horavisitaFitxavisita} referencia HORAVISITA (idHoravisita),
 - on {accesFitxavisita} referencia ACCES (idAcces)
- **PARAMETRE** (idParametre, nomParametre, valorParametre)
- **LOG** (idLog, procesLog, datahoraLog, entradaLog, sortidaLog, rspLog)

2.5.3. Observacions

Cal fer menció especial a les taules PARAMETRE i LOG, les quals no formen part del diagrama E / R, perquè no presenten cap relació en concret:

PARAMETRES:

Específicament el projecte requereix una taula que contingui unes dades anomenades Paràmetres de sistema, aquestes dades serviran per assignar valors per defecte a alguns atributs com per exemple el temps mínim de visita.

LOG:

Totes les crides a procediment quedaran registrades en una taula per a la seva posterior consulta. En acabar l'execució del procediment obtindrem les següents dades:

- Nom del procediment.
- Paràmetres d'entrada.
- Paràmetres de sortida.
- Valor RSP.
- Data i hora.

Trobem una dada anomenada Valor RSP, que definirem a continuació segons l'enunciat del projecte:

Valor RSP:

Tot procediment portarà un indicador de realització correcte o incorrecte, l'RSP constarà d'un paràmetre de sortida de tipus string que mostrarà el valor 'O.K.', en cas de que l'execució hagi finalitzar amb èxit o 'Error + tipus d'error' en cas d'error. Tipus d'error ens vindrà donat pel tractament de l'excepció que ja haurem fet.

3. Programació de la base de dades (Model físic)

A la fase de disseny físic s'especifiquen les estructures on s'han d'emmagatzemar les dades i l'organització de l'arxiu de la base de dades. També s'implementen els programes d'aplicació per les transaccions i els procediments propis del sistema.

3.1. Creació dels Scripts (Procediments)

Primer programarem els Scripts de creació de les taules que contindran les dades, després tots els procediments d'accés a aquestes dades.

3.1.1. Programari utilitzat

Oracle és sens dubte una de les millors bases de dades que tenim en el mercat, te moltes característiques que ens garanteixen la seguretat i integritat de les dades; que les transaccions s'efectuaran de manera correcta, sense causar inconsistència; desenvolupament de la capa de dades utilitzant: procediments, funcions, desencadenants i paquets; i el processament de grans volums d'informació estarà també assegurada. Per a la realització d'aquest projecte utilitzem la versió: **Oracle9i Enterprise Edition Release 9.0.1.1.1**.

Per facilitar els accessos a la base de dades utilitzem el **PL/SQL Developer 6.0.0.840**, que és un IDE específicament dirigit al desenvolupament d'unitats de programes emmagatzemats per a bases de dades d'Oracle. Algunes de les principals característiques d'aquest software que s'han utilitzat en el desenvolupament d'aquest projecte son:

Editor de PL/SQL: Ressalta la sintaxis, ajuda SQL i PL/SQL, descripció d'objectes, assistent de codi, consells de compilador, amb una gran facilitat d'us acompanyat de botons i vincles.

Depurador de codi: Ofereix depuració pas a pas per instruccions, per procediments, establir punts de ruptura, veure i establir valors de variables, veure pila de crides, etc. Es pot depurar qualsevol tipus d'unitat de programa sense fer-li cap modificació.

Finestra SQL: Aquesta finestra permet introduir qualssevol sentència SQL i veure o editar els resultats en una graella que suporta el mode "Query by example", per a buscar registres específics en un conjunt de dades. Hi ha un registre de sentències executades fàcilment recuperables.

Finestra de comandes: Per a desenvolupar i executar "scripts" SQL utilitzem aquesta finestra, te el mateix aspecte i funcionament de SQL*Plus i a més presenta un editor integrat ressaltant el codi.

Finestra de Test: Permet provar el funcionament dels procediments i funcions amb paràmetres molt fàcilment, cal introduir les dades en una graella i executar el procediment, si és necessari, pas a pas. A la mateixa finestra es pot observar els paràmetres de sortida.

3.1.2. Seqüències SQL de creació de les taules

Definició de les taules, columnes amb els tipus de dades i claus principals. No es definiran claus foranes, aquest tipus d'integritat s'implementarà en els procediments, d'aquesta manera controlem les excepcions i podem personalitzar-les i detallar-les a la nostra mida.

Durant l'execució dels procediments de manipulació, creació o eliminació de dades es tindrà cura d'avaluar tots els impactes a la resta de les taules. Cal fer menció al procés d'eliminació de dades, tampoc s'ha usat el "Delete on Cascade", que elimina automàticament totes les referències a la clau principal eliminada. El mateix procediment d'eliminació s'encarregarà de cercar totes les dades vinculades i esborrar-les sense malmetre la integritat de la base de dades.

S'ha utilitzat les els següents tipus de dades:

VARCHAR2: Aquest tipus de dades permet una gran flexibilitat en el seu tractament.

INTEGER: Tipus numèric que permet guardar valors compresos entre el rang:

-2147482647 i +2147482647.

Date: Permet guardar informació sobre la data: dia, mes, any, hora, minut. Amb el format 'DD-MM-YY HH24:MI'. El pes d'una dada Date és de set bytes.

3.1.2.1. Taula PERSONA

```
CREATE TABLE PERSONA (
  idPersona VARCHAR2 (10) NOT NULL,
  nomPersona VARCHAR2 (15) NOT NULL,
  cognom1Persona VARCHAR2 (15) NOT NULL,
  cognom2Persona VARCHAR2 (15) NOT NULL,
  datanaixementPersona DATE NOT NULL,
  nifPersona VARCHAR2 (9) NOT NULL,
  carrerPersona VARCHAR2 (20) NOT NULL,
  numeroPersona INT NOT NULL,
  pisPersona INT,
  portaPersona VARCHAR2 (4),
  cpPersona INT NOT NULL,
  poblacioPersona VARCHAR2 (20) NOT NULL,
  telefonPersona VARCHAR2 (10),
  PRIMARY KEY (idPersona)
);
```

Conté la informació comú que parteix la generalització de les entitats: Pacient, Metge, ATS i Administratiu. Tots els camps no poden contenir valors nuls excepte els que guarden la informació del pis, porta i telèfon. La clau principal serà la identificació de la persona.

3.1.2.2. Taula MEDICAMENT

```
CREATE TABLE MEDICAMENT (
  refMedicament VARCHAR2 (15) NOT NULL,
  nomMedicament VARCHAR2 (20) NOT NULL,
  fabricantMedicament VARCHAR2 (20) NOT NULL,
  familiaMedicament VARCHAR2 (20) NOT NULL,
  indicacionsMedicament VARCHAR2 (30),
  contradiccionsMedicament VARCHAR2 (30),
  administracioMedicament VARCHAR2 (30),
  PRIMARY KEY (refMedicament)
);
```

Descripció d'un medicament, només els camps indicacions, contradiccions i administració poden ser nuls. La clau principal serà la referència del medicament.

3.1.2.3. Taula ESPECIALITAT

```
CREATE TABLE ESPECIALITAT (
  idEspecialitat VARCHAR2 (10) NOT NULL,
  nomEspecialitat VARCHAR2 (20) NOT NULL,
  PRIMARY KEY (idEspecialitat)
);
```

Especialitat que pot tenir un metge segons la seva preparació. No pot haver-hi camps sense valors. Un identificador de cada especialitat serà la clau principal.

3.1.2.4. Taula ESTAT

```
CREATE TABLE ESTAT (  
  idEstat VARCHAR2 (10) NOT NULL,  
  nomEstat VARCHAR2 (20) NOT NULL,  
  PRIMARY KEY (idEstat)  
);
```

Conté els estats laborals que pot tenir el personal del CM: Actiu, inactiu, baixa, vacances.... No pot contenir valors nuls i la clau principal és un identificador per a cada estat.

3.1.2.5. Taula ACCES

```
CREATE TABLE ACCES (  
  idAcces VARCHAR2 (10) NOT NULL,  
  tipusAcces VARCHAR2 (20) NOT NULL,  
  PRIMARY KEY (idAcces)  
);
```

Bàsicament conté les dues maneres per les que podem accedir al CM: concertat i urgència. La clau principal és un identificador del tipus d'accés i els camps no poden contenir valors nuls.

3.1.2.6. Taula TORN

```
CREATE TABLE TORN (  
  idTorn VARCHAR2 (10) NOT NULL,  
  nomTorn VARCHAR2 (20) NOT NULL,  
  PRIMARY KEY (idTorn)  
);
```

Defineix els noms dels torns que pot treballar el personal del centre mèdic. Un identificador per cada torn és la clau principal. No pot contenir cap camp valors nuls.

3.1.2.7. Taula DIASETMANA

```
CREATE TABLE DIASETMANA (  
  idDiasetmana VARCHAR2 (10) NOT NULL,  
  nomDiasetmana VARCHAR2 (10) NOT NULL,  
  PRIMARY KEY (idDiasetmana)  
);
```

Defineix els dies de la setmana, els camps no poden contenir valors nuls i l'identificador de cada dia és la clau principal.

3.1.2.8. Taula RECEPTA

```
CREATE TABLE RECEPTA (
  idRecepta INT NOT NULL,
  fitxavisitaRecepta VARCHAR2 (10) NOT NULL,
  medicamentRecepta VARCHAR2 (15) NOT NULL,
  PRIMARY KEY (idRecepta)
);
```

Conté la relació de medicaments receptats pel metge al pacient durant la visita. No pot contenir valors nuls. La clau principal és l'identificador de la recepta. Caldria una clau forana que fes referència a la taula MEDICAMENT, però com hem dit abans, per procediment controlarem que el medicament que formi part de la recepta, existeixi a la taula.

3.1.2.9. Taula PACIENT

```
CREATE TABLE PACIENT (
  idPacient VARCHAR2 (10) NOT NULL,
  metgeAssignat VARCHAR2 (10) NOT NULL,
  PRIMARY KEY (idPacient)
);
```

Generalització / Especialització de Persona, conté l'atribut diferenciador del metge assignat. Aquest atribut és una clau forana cap a l'identificador de Metge, però ho declararem més endavant quan tinguem la taula Metge creada. Declarem la identificació del pacient com a clau principal però també caldria fer-ho com a forana, fent referència a la taula PERSONA, però com hem dit abans, per procediment controlarem que la persona que sigui pacient, existeixi a la taula PERSONA, també ens ocuparem que l'eliminació del registre a la taula PERSONA elimini el registre corresponent a la taula PACIENT. Els camps no poden tenir valors nuls.

3.1.2.10. Taula METGE

```
CREATE TABLE METGE (
  idMetge VARCHAR2 (10) NOT NULL,
  especialitatMetge VARCHAR2 (10) NOT NULL,
  estatMetge VARCHAR2 (10) NOT NULL,
  atsMetge VARCHAR2 (10) NOT NULL,
  PRIMARY KEY (idMetge)
);
```

Generalització / Especialització de Persona, conté els atributs diferenciadors d'especialitat, estat i ats associat. Aquests atributs haurien de ser claus foranes cap a les taules corresponents, però per procediment ens assegurarem que cada element al qual fem referència, pertanyi a les taules principals. Declarem la identificació del metge com a clau principal però també ho faríem com a forana, però com hem dit abans, per procediment controlarem que la persona que sigui metge, existeixi a la taula PERSONA, també ens ocuparem que l'eliminació del registre a la taula PERSONA elimini el registre corresponent a la taula METGE. Els camps no poden tenir valors nuls.

3.1.2.11. Taula ATS

```
CREATE TABLE ATS (
  idAts VARCHAR2 (10) NOT NULL,
  estatAts VARCHAR2 (10) NOT NULL,
  metgeAts VARCHAR2 (10) NOT NULL,
  PRIMARY KEY (idAts)
);
```

Generalització / Especialització de Persona, conté els atributs diferenciadors d'estat i metge associat. Aquests atributs serien claus foranes cap a les taules corresponents però per procediment ens assegurarem que cada element al qual fem referència, pertanyi a les taules principals. Declarem la identificació del ats com a clau principal també ho fariem com a forana, però com hem dit abans, per procediment controlarem que la persona que sigui ATS, existeixi a la taula PERSONA, també ens ocuparem que l'eliminació del registre a la taula PERSONA elimini el registre corresponent a la taula ATS. Els camps no poden tenir valors nuls.

3.1.2.12. Taula ADMINISTRATIU

```
CREATE TABLE ADMINISTRATIU (
  idAdministratiu VARCHAR2 (10) NOT NULL,
  estatAdministratiu VARCHAR2 (10) NOT NULL,
  PRIMARY KEY (idAdministratiu)
);
```

Generalització / Especialització de Persona, conté l'atribut diferenciador de l'estat. Aquest atribut seria clau forana cap a les taula corresponent però per procediment ens assegurarem que cada element al qual fem referència, pertanyi a les taula principal. Declarem la identificació de l'administratiu com a clau principal també ho fariem com a forana, però com hem dit abans, per procediment controlarem que la persona que sigui administratiu, existeixi a la taula PERSONA, també ens ocuparem que l'eliminació del registre a la taula PERSONA elimini el registre corresponent a la taula ADMINISTRATIU. Els camps no poden tenir valors nuls.

3.1.2.13. Taula HORAVISITA

```
CREATE TABLE HORAVISITA (
  idHoravisita VARCHAR2 (10) NOT NULL,
  datahorainiciHoravisita DATE NOT NULL,
  datahorafiHoravisita DATE NOT NULL,
  metgeHoravisita VARCHAR2 (10) NOT NULL,
  pacientHoravisita VARCHAR2 (10) NOT NULL,
  administratiuHoravisita VARCHAR2 (10) NOT NULL,
  accesHoravisita VARCHAR2 (10) NOT NULL,
  observacionsHoravisita VARCHAR2 (30),
  PRIMARY KEY (idHoravisita)
);
```

Conté la informació prèvia a la visita, només el camp observacions pot ser nul. El camp Date tindrà un tractament especial per inserir el format correcte. La clau principal és un identificador per cada Horavisita. Pacient i metge serien claus foranes cap a les taules corresponents però per procediment ens assegurarem que al fer la inserció, cada element al qual fem referència, pertanyi a les taules principals. Una eliminació d'aquests elements a la taula principal, no afecta a aquesta taula, ja que la informació a HORAVISITA ha de perdurar en el temps.

3.1.2.14. Taula HORARILABORAL

```
CREATE TABLE HORARILABORAL (
  idHorarilaboral INT NOT NULL,
  personaHorarilaboral VARCHAR2 (10) NOT NULL,
  diaHorarilaboral VARCHAR2 (10) NOT NULL,
  tornHorarilaboral VARCHAR2 (10) NOT NULL,
  guardiaHorarilaboral INT NOT NULL,
  PRIMARY KEY (idHorarilaboral)
);
```

Taula de consulta dels horaris de treball del personal del centre mèdic, tan sols el camp guàrdia pot contenir valor nul. Si un membre del CM deixa el seu lloc de treball, no apareixerà a la taula, el mateix passa si un dia de la setmana deixa de ser laboral o desapareix un torn. La clau principal és un identificador de cada registre.

3.1.2.15. Taula FITXAVISITA

```
CREATE TABLE FITXAVISITA (
  idFitxavisita VARCHAR2 (10) NOT NULL,
  pacientFitxavisita VARCHAR2 (10) NOT NULL,
  metgeFitxavisita VARCHAR2 (10) NOT NULL,
  atsFitxavisita VARCHAR2 (10) NOT NULL,
  horavisitaFitxavisita VARCHAR2 (10) NOT NULL,
  datahorainiciFitxavisita DATE NOT NULL,
  datahorafiFitxavisita DATE NOT NULL,
  diagnosticFitxavisita VARCHAR2 (300),
  observacionsFitxavisita VARCHAR2 (300),
  PRIMARY KEY (idFitxavisita)
);
```

Guarda el resultat final del pas del pacient pel centre mèdic, només el diagnòstic, les observacions i la recepta poden ser nuls. Els camps Date tindran un tractament especial per inserir el format correcte Un identificador de cada fitxa de visita serà la clau principal. Pacient, metge, ATS, hora visita, serien claus foranes cap a les taules corresponents però per procediment ens assegurarem que cada element al qual fem referència, pertanyi a les taules principals. Aquests registres seran persistents en el temps, ja que el grup de fitxes de visita d'un determinat pacient, suposarà el seu historial mèdic i la informació continguda pot interessar, encara, després de la mort del pacient.

3.1.2.16. Taula PARAMETRE

```
CREATE TABLE PARAMETRE (
  idParametre VARCHAR2 (10) NOT NULL,
  nomParametre VARCHAR2 (20) NOT NULL,
  valorParametre VARCHAR2 (20) NOT NULL,
  PRIMARY KEY (idParametre)
);
```

Conté els paràmetres prefixats necessaris pel funcionament del centre mèdic. Un identificador per cada paràmetre és la clau principal.

3.1.2.17. Taula REGISTRELOG

```
CREATE TABLE REGISTRELOG (
  idLog VARCHAR2 (10) NOT NULL,
  procesLog VARCHAR2 (30) NOT NULL,
  datahoraLog DATE NOT NULL,
  entradaLog VARCHAR2 (300) NOT NULL,
  sortidaLog VARCHAR2 (300) NOT NULL,
  rspLog VARCHAR2 (300) NOT NULL,
  PRIMARY KEY (idLog)
);
```

Taula que registra tota l'activitat de la base de dades, recull els paràmetres d'entrada, els de sortida i la informació de la correcta realització. Un identificador de cada activitat és la clau principal.

3.1.2.18. Seqüències SQL per omplir taules

Introduïrem les dades estàtiques, que han d'estar presents a les taules pel correcte funcionament de la base de dades.

3.1.2.18.1.1. Dades per ESTAT

```
INSERT INTO ESTAT VALUES ('1', 'Actiu');
INSERT INTO ESTAT VALUES ('2', 'Inactiu');
INSERT INTO ESTAT VALUES ('3', 'Baixa');
INSERT INTO ESTAT VALUES ('4', 'Vacances');
```

Cal inserir els estats en que es pot trobar l'empleat, ja que durant el procés d'introducció de dades s'ha d'especificar.

3.1.2.18.1.2. Dades per ACCES

```
INSERT INTO ACCES VALUES ('1', 'Concertat');
INSERT INTO ACCES VALUES ('2', 'Urgència');
```

Definim els dos tipus d'accés al CM.

3.1.2.18.1.3. Dades per TORN

```
INSERT INTO TORN VALUES ('1', 'Matí');
INSERT INTO TORN VALUES ('2', 'Tarda');
```

Definim el nom dels torns en que poden treballar els empleats del CM.

3.1.2.18.1.4. Dades per DIASETMANA

```
INSERT INTO DIASETMANA VALUES ('1', 'Dilluns');
INSERT INTO DIASETMANA VALUES ('2', 'Dimarts');
INSERT INTO DIASETMANA VALUES ('3', 'Dimecres');
INSERT INTO DIASETMANA VALUES ('4', 'Dijous');
INSERT INTO DIASETMANA VALUES ('5', 'Divendres');
INSERT INTO DIASETMANA VALUES ('6', 'Dissabte');
INSERT INTO DIASETMANA VALUES ('7', 'Diumenge');
```

Seràn necessaris per introduir dades a la taula Horarilaboral.

3.1.2.18.1.5. Dades per PARAMETRE

```
INSERT INTO PARAMETRE VALUES  
( '1', 'Durada Visita', '30');
```

A la taula paràmetres hi haurà una sèrie de valors predefinitos que serviran per les tasques quotidianes del CM.

3.1.2.18.1.6. Dades per METGE i ATS

```
INSERT INTO METGE VALUES  
( '00000000', '1', '1', '00000000');  
INSERT INTO ATS VALUES  
( '00000000', '1', '00000000');
```

Necessitem inserir aquestes dades a les taules METGE i ATS per utilitzar els procediments, ja que quan inserim el primer metge és necessari assignar-li un ats i a l'inrevés. Farem servir aquestes dades 'fantasma' per realitzar-ho. Posteriorment els registres que continguin aquestes dades podran ser modificats.

3.1.3. Sentències SQL de definició de Funcions

Definim les funcions d'accés a la base de dades per suport als procediments. Per motius d'espai no s'inclou el codi de les funcions en aquest document, el trobarem a "TFC-Consultori.sql". Aquí només es descriuran els passos principals que s'hi duen a terme.

3.1.3.1. NOHIES

Funció que utilitzem per comprovar si existeix un registre en una taula determinada. Passem per paràmetres: nom de la taula, nom de la columna i valor del registre, mitjançant una consulta SQL comprovem si el valor hi és, en cas afirmatiu, la funció retorna el valor fals, en cas contrari retorna el valor vertader.

Utilitzem aquesta funció en gairebé tots els procediments, ja que és molt útil per relacionar claus foranes amb principals. Exemple: si volem crear un metge, primer hauré de comprovar que existeix a la taula persona, aquesta funció ens fa la feina.

3.1.3.2. TEMPSVISITACORRECTE

Retorna cert si l'espai temporal entre les dues hores passades per paràmetre supera el valor en minuts indicat a la taula PARAMETRE.

L'utilitzem per inserir una hora de visita. Quan el personal administratiu vol introduir una nova hora de visita, ha d'indicar les hores previstes d'inici i de fi, després es comprova que la duració de la visita superi el temps mínim, si és així ens retorna cert.

3.1.3.3. METGEDISPONIBLE

Retorna cert si el metge està disponible en les hores previstes de visita. El personal administratiu quan insereix una hora de visita, indica l'hora d'inici i el metge que ha d'intervenir, després es comprova a la taula HORARILABORAL que el metge realitzi el torn de treball corresponent al indicat per l'hora d'inici de visita.

3.1.3.4. ELIMINARRELACIONS

Funció que s'utilitza per donar suport a determinats procediments d'eliminació de registres. Quan es produeix l'eliminació d'un registre determinat, que afecta a altres taules, aquesta funció substitueix el valor que fa referència en concepte de clau forana al registre eliminat pel valor "0". Retorna cert si la cerca i substitució dels registres amb referència al registre eliminat, s'ha produït sense problemes. Al mateix temps també utilitza la funció NOHIES per fer les cerques a les taules afectades.

3.1.4. Sentències SQL de definició de Procediments

Definim els procediments d'accés a les taules de la base de dades: Inserir, eliminar, modificar i altres més complexos. Per motius d'espai no s'inclou el codi dels Procediments en aquest document, el trobarem a "TFC-Consultori.sql". Aquí només es descriuran els passos principals que s'hi duen a terme.

3.1.4.1. REGISTRARLOG

Procediment que es crida quan es produeix un accés a la base de dades per part dels procediments o funcions, la seva tasca és la de registrar l'acció realitzada, tan si s'ha realitzat correctament o hi ha hagut algun error durant el procés, en aquest cas registra la informació de l'errada produïda. Insereix les dades: nom del procés, hora, dades d'entrada, sortida i valor RSP passats per paràmetre, a la taula REGISTRELOG.

3.1.4.2. CREARPERSONA

Procediment que registra una persona a la taula PERSONA amb tots els seus atributs, nom, cognom, adreça..., passats per paràmetres.

Comprovacions prèvies:

- No existeix la persona a la taula PERSONA.

Resultat:

- Persona inserida a la taula PERSONA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.3. MODIFICARPERSONA

Procediment que modifica una persona a la taula PERSONA amb tots els seus atributs, nom, cognom, adreça..., passats per paràmetres.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.

Resultat:

- Persona modificada a la taula PERSONA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.4. ELIMINARPERSONA

Procediment que elimina una persona de la taula PERSONA amb la seva identificació passada per paràmetre.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.

Resultat:

- Persona eliminada a la taula PERSONA.
- SI es dona el cas, pacient eliminat de la taula PACIENT.
- SI es dona el cas, metge eliminat de la taula METGE.
- SI es dona el cas, ats eliminat de la taula ATS.
- SI es dona el cas, Administratiu eliminat de la taula ADMINISTRATIU.
- SI es dona el cas, ATS assignat amb valor "0" a la taula METGE.
- SI es dona el cas, metge assignat amb valor "0" a la taula PACIENT.
- SI es dona el cas, metge assignat amb valor "0" a la taula ATS.
- SI es dona el cas, horari laboral que fa referència a la persona, esborrat.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.5. CREARPACIENT

Procediment que crea un pacient amb el metge assignat passats per referència.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.

Resultat:

- Pacient creat a la taula PACIENT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.6. MODIFICARPACIENT

Procediment que modifica un pacient amb el metge assignat passats per referència.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.

Resultat:

- Pacient modificat a la taula PACIENT.

- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.7. CREARMETGE

Procediment que crea un metge, especificant el seu estat, l'especialitat i l'ATS assignat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- No existeix el metge a la taula METGE.
- Existeix l'estat a la taula ESTAT.
- Existeix l'especialitat a la taula ESPECIALITAT.
- Existeix l'ats a la taula ATS.

Resultat:

- Metge creat a la taula METGE.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.8. MODIFICARMETGE

Procediment que modifica un metge, especificant la identitat, l'estat, l'especialitat i l'ATS assignat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.
- Existeix l'estat a la taula ESTAT.
- Existeix l'especialitat a la taula ESPECIALITAT.
- Existeix l'ats a la taula ATS.

Resultat:

- Metge modificat a la taula METGE.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.9. CREAR ATS

Procediment que crea un ats, especificant la identitat, l'estat i el metge assignat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- No existeix l'ats a la taula ATS.
- Existeix l'estat a la taula ESTAT.
- Existeix el metge a la taula METGE.

Resultat:

- Ats creat a la taula ATS.

- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.10. MODIFICARATS

Procediment que modifica un ats, especificant la identitat, l'estat i el metge assignat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix l'ats a la taula ATS.
- Existeix l'estat a la taula ESTAT.
- Existeix el metge a la taula METGE.

Resultat:

- Ats modificat a la taula ATS.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.11. CREAMADMINISTRATIU

Procediment que crea un administratiu, especificant la identitat i l'estat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- No existeix l'administratiu a la taula ADMINISTRATIU.
- Existeix l'estat a la taula ESTAT.

Resultat:

- Administratiu creat a la taula ADMINISTRATIU.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.12. MODIFICARADMINISTRATIU

Procediment que modifica un administratiu i el seu estat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix l'administratiu a la taula ADMINISTRATIU.
- Existeix l'estat a la taula ESTAT.

Resultat:

- Administratiu modificat a la taula ADMINISTRATIU.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.13. CREARMEDICAMENT

Procediment que crea un medicament, especificant la referència, el nom, el fabricant, família, indicacions, contradiccions i administració.

Comprovacions prèvies:

- No existeix el medicament a la taula MEDICAMENT.

Resultat:

- Medicament creat a la taula MEDICAMENT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.14. MODIFICARMEDICAMENT

Procediment que modifica un medicament, especificant la referència, el nom, el fabricant, família, indicacions, contradiccions i administració.

Comprovacions prèvies:

- Existeix el medicament a la taula MEDICAMENT.

Resultat:

- Medicament modificat a la taula MEDICAMENT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.15. ELIMINARMEDICAMENT

Procediment que elimina un medicament, especificant la referència.

Comprovacions prèvies:

- Existeix el medicament a la taula MEDICAMENT.

Resultat:

- Medicament eliminat a la taula MEDICAMENT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.16. CREARESPECIALITAT

Procediment que crea una especialitat mèdica, especificant la identificació i el nom.

Comprovacions prèvies:

- No existeix l'especialitat a la taula ESPECIALITAT.

Resultat:

- Especialitat creada a la taula ESPECIALITAT.

- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.17. MODIFICARESPECIALITAT

Procediment que modifica una especialitat mèdica, especificant la identificació i el nom.

Comprovacions prèvies:

- Existeix l'especialitat a la taula ESPECIALITAT.

Resultat:

- Especialitat modificada a la taula ESPECIALITAT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.18. ELIMINARESPECIALITAT

Procediment que modifica una especialitat mèdica, especificant la identificació.

Comprovacions prèvies:

- Existeix l'especialitat a la taula ESPECIALITAT.

Resultat:

- Especialitat eliminada a la taula ESPECIALITAT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.19. CREAMACCES

Procediment que crea un tipus d'accés, especificant la identificació i el nom.

Comprovacions prèvies:

- No existeix l'accés a la taula ACCES.

Resultat:

- Accés creat a la taula ACCES.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.20. MODIFICARACCES

Procediment que modifica un tipus d'accés, especificant la identificació i el nom.

Comprovacions prèvies:

- Existeix l'accés a la taula ACCES.

Resultat:

- Accés modificat a la taula ACCES.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.21. ELIMINARACCES

Procediment que elimina un tipus d'accés, especificant la identificació.

Comprovacions prèvies:

- Existeix l'accés a la taula ACCES.

Resultat:

- Accés eliminat a la taula ACCES.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.22. CREARTORN

Procediment que crea una torn, especificant la identificació i el nom.

Comprovacions prèvies:

- No existeix el torn a la taula TORN.

Resultat:

- Torn creat a la taula TORN.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.23. MODIFICARTORN

Procediment que modifica un torn, especificant la identificació i el nom.

Comprovacions prèvies:

- Existeix el torn a la taula TORN.

Resultat:

- Torn modificat a la taula TORN.

- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.24. ELIMINARTORN

Procediment que elimina un torn, especificant la identificació i el nom.

Comprovacions prèvies:

- Existeix el torn a la taula TORN.

Resultat:

- Torn eliminat a la taula TORN.
- Si es dona el cas, torn horari laboral amb valor "0" a la taula HORARILABORAL.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.25. CREARESTAT

Procediment que crea un estat, especificant la identificació i el nom.

Comprovacions prèvies:

- No existeix l'estat a la taula ESTAT.

Resultat:

- Estat creat a la taula ESTAT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.26. MODIFICARESTAT

Procediment que modifica un estat, especificant la identificació i el nom per paràmetres.

Comprovacions prèvies:

- Existeix l'estat a la taula ESTAT.

Resultat:

- Estat modificat a la taula ESTAT.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.27. ELIMINARESTAT

Procediment que elimina un estat, especificant la identificació per paràmetres

Comprovacions prèvies:

- Existeix l'estat a la taula ESTAT.

Resultat:

- Estat eliminat a la taula ESTAT.
- SI es dona el cas, estat del metge amb valor "0" a la taula METGE.
- SI es dona el cas, estat de l'ats amb valor "0" a la taula ATS.
- SI es dona el cas, estat de l'administratiu amb valor "0" a la taula ADMINISTRATIU.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.28. INSERIRMEDICAMENTRECEPTA

Procediment que afegeix un medicament a una recepta, que pertany a una fitxa visita. Passem per paràmetres la identificació del medicament i la identificació de la fitxa visita a la qual pertany la recepta.

Comprovacions prèvies:

- Existeix el medicament a la taula MEDICAMENT.
- Existeix la fitxa de visita.
- No existeix el medicament a la taula RECEPTA

Resultat:

- Medicament inserit a la taula RECEPTA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.29. ELIMINAMEDICAMENTRECEPTA

Procediment que elimina un medicament d'una recepta, que pertany a una fitxa visita. Passem per paràmetres la identificació del medicament i la identificació de la fitxa visita a la qual pertany la recepta.

Comprovacions prèvies:

- Existeix el medicament a la taula MEDICAMENT.
- Existeix el medicament a la taula RECEPTA

Resultat:

- Medicament eliminat de la taula RECEPTA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.30. MODIFICARHORARILABORAL

Procediment que crea o modifica un horari laboral. L'horari laboral del personal del CM és a una setmana vista, tot el personal del centre mèdic tindrà definit els torns de treball de tots els dies de la setmana, inclòs les guàrdies si s'escau, ja que els torns poden ser rotatius o no. Es passa per paràmetres la persona, el dia de la setmana i el torn que realitzarà. Si no s'ha definit el torn de

treball d'aquest dia en concret per aquesta persona, es crea, si aquesta persona te definit un altre torn per aquest dia es modifica.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el dia a la taula DIASETMANA.
- Existeix el torn a la taula TORN.

Resultat:

- Horari laboral per una persona i dia en concret definit a la taula HORARILABORAL.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.31. ELIMINARHORARILABORAL

Procediment que elimina un horari laboral (dia, persona i torn). Es passa per paràmetres la persona i el dia de la setmana. Si no s'ha definit el torn de treball d'aquest dia en concret per aquesta persona sorgeix l'excepció.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el dia a la taula DIASETMANA.
- Existeix horari laboral a la taula HORARILABORAL.

Resultat:

- Horari laboral per una persona i dia en concret eliminat a la taula HORARILABORAL.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.32. CREARGUARDIA

Procediment que assigna un torn de guàrdia. Posa a "1" l'atribut guàrdia del registre on defineix el torn diari de la persona a HORARILABORAL. Es passen per paràmetres la identificació de la persona i el dia de la setmana afectat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el dia a la taula DIASETMANA.
- Existeix horari laboral a la taula HORARILABORAL.

Resultat:

- Guàrdia assignada a una persona en un torn i dia concret a la taula HORARILABORAL.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.33. ELIMINARGUARDIA

Procediment que lliura un torn de guàrdia. Posa a "0" l'atribut guàrdia del registre on defineix el torn diari de la persona a HORARILABORAL. Es passen per paràmetres la identificació de la persona i el dia de la setmana afectat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el dia a la taula DIASETMANA.
- Existeix horari laboral a la taula HORARILABORAL.

Resultat:

- Guàrdia no assignada a una persona en un torn i dia concret a la taula HORARILABORAL.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.34. CREAMHORAVISITA

Procediment en el qual el personal administratiu crea una hora de visita. A partir d'unes hores d'inici i fi previstes, es comprova la disponibilitat del metge i es confirma l'hora de visita. Es passen per paràmetres: l'hora d'inici, de fi, el nom del pacient, del metge i les observacions addicionals a tenir en compte.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.
- L'espai de temps entre l'hora d'inici i hora de fi és superior al que es marca a la taula PARAMETRES.
- El metge està disponible per a realitzar la visita, que sigui dia laborable per ell i que no tingui una cita assignada en aquest espai de temps.

Resultat:

- Hora de visita acordada entre metge i pacient creada a HORAVISITA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.35. MODIFICARHORAVISITA

Procediment en el qual el personal administratiu modifica una hora de visita. Com que tots els elements son susceptibles a canvi s'actua com si es tractés d'una petició d'hora de visita nova, per tant a partir d'unes hores d'inici i fi previstes, es comprova la disponibilitat del metge i es confirma l'hora de visita. Es passen per paràmetres: l'hora d'inici, de fi, el nom del pacient, del metge i les observacions addicionals a tenir en compte. Abans salvem l'hora visita original que es restaura en cas d'alguna incompatibilitat.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.

- L'espai de temps entre l'hora d'inici i hora de fi és superior al que es marca a la taula PARAMETRES.
- El metge està disponible per a realitzar la visita, que sigui dia laborable per ell i que no tingui una cita assignada en aquest espai de temps.

Resultat:

- Hora de visita acordada entre metge i pacient modificada a HORAVISITA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.36. ELIMINARHORAVISITA

Procediment en el qual el personal administratiu anul·la una hora de visita. A partir de la hora d'inici i el metge s'anul·la, quedant disponible aquest espai de temps per una altre visita . Es passen per paràmetres: l'hora d'inici i el nom del metge.

Comprovacions prèvies:

- Existeix el metge a la taula METGE.
- Hora visita assignada prèviament a la taula HORAVISITA.

Resultat:

- Hora de visita acordada entre metge i pacient anul·lada a HORAVISITA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.37. CREAMFITXAVISITA

Procediment que registra el resultat de la visita entre metge i pacient. Queda una fitxa visita relacionada amb la hora visita prèvia i amb un llistat de medicaments receptats. Es passen per paràmetres: el pacient, el metge i l'ats com a personal humà implicat, la reserva de visita prèvia, la hora d'inici, la hora de fi, el diagnòstic i les observacions pertinents.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.
- Existeix l'ats a la taula ATS.
- Hora visita assignada prèviament a la taula HORAVISITA.

Resultat:

- Fitxa resultant de la visita registrada a FITXAVISITA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.38. MODIFICARFITXAVISITA

Procediment que modifica una fitxa de visita ja registrada. Es passen per paràmetres: la identificació de fitxa visita, el pacient, el metge i l'ats, la reserva de visita prèvia, la hora d'inici, la hora de fi, el diagnòstic i les observacions pertinents.

Comprovacions prèvies:

- Existeix la persona a la taula PERSONA.
- Existeix el metge a la taula METGE.
- Existeix l'ats a la taula ATS.
- Hora visita assignada prèviament a la taula HORAVISITA.
- Existeix la fitxa visita a la taula FITXAVISITA.

Resultat:

- Fitxa resultant de la visita modificada a FITXAVISITA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

3.1.4.39. ELIMINARFITXAVISITA

Procediment que elimina una fitxa de visita, es passa per paràmetre: la identificació de la fitxa de visita.

Comprovacions prèvies:

- Existeix la fitxa visita a la taula FITXAVISITA.

Resultat:

- Fitxa resultant de la visita eliminada a FITXAVISITA.
- Recepta eliminada a la taula RECEPTA.
- Registre a REGISTRELOG amb tots els detalls de l'execució del procediment.

4. Proves a la base de dades

Per a la realització de les proves, s'ha utilitzat l'aplicació Test del PL/SQL developer, que et permet provar procediment per procediment, introduint els paràmetres d'entrada i observar els paràmetres de sortida amb molta facilitat. També permet depurar el codi i executar pas a pas per determinar si es dona el cas alguna errada.

S'ha realitzat proves molt exhaustives a tots els procediments, un total del trenta nou, totes les proves s'han realitzat correctament. Per no allargar massa el contingut del document, s'han omès les proves als procediments menys complexos, ens referim als que realitzen les tasques d'inserció, modificació i eliminació d'elements de les taules, als quals només és necessari una simple sentència SQL per a realitzar l'acció.

Centrarem les proves als procediments més complexos, els que utilitzen les funcions descrites abans, els procediments son: CREAMHORAVISITA, MODIFICARHORAVISITA i ELIMINARPERSONA.

Utilitzarem una taula on s'indicarà:

- Paràmetres d'entrada.
- Resultat esperat.
- Valor RSP.

També, en cas d'aparèixer una Excepció, s'informarà a quina funció o procediment es produeix.

Disposarem de les següents dades a les taules per realitzar les proves, a part de les que s'han introduït posteriorment de la creació de les taules (Torns, dies de la setmana...).

4.1. Inserció de dades

Hi haurà un mínim de dades per a la realització de les proves, només tindrem dades per a la realització de les proves dels procediments complexos descrits anteriorment. Inserirem les dades a les següents taules:

4.1.1. PERSONA

```

exec crearpersona ('12345621', 'Jaume', 'Refeques', 'Antolí', '14-12-1980', '12345621', 'Tinet',
45, null,null, 17403, 'Sant Hilari', 677784536);

exec crearpersona ('12345678', 'Josep', 'Gonzalez', 'Pagès', '21/03/1972', '12345678', 'Est', 22,
3, '2-b', 17403, 'Sant Hilari', 972456654);

exec crearpersona ('12345610', 'Ramon', 'Pallarols', 'Masaguer', '11/09/1982', '12345610', 'Oest',
6, 7,null, 17403, 'Sant Hilari', 698765434);

exec crearpersona ('12345679', 'Antoni', 'Perez', 'Ramirez', '19/05/1970', '12345679', 'Nord', 2,
5, '4-c', 17403, 'Sant Hilari', 972456680);

exec crearpersona ('12345613', 'Carme', 'Peris', 'Sucarrat', '10/03/1969', '12345613', 'SudEst',
11, 4, '3-4', 17403, 'Sant Hilari', 688763234);

exec crearpersona ('12345612', 'Maria', 'Gutiérrez', 'Cases', '11/07/1955', '12345612', 'NordOest',
100, 34, '1-b', 17403, 'Sant Hilari', 972567732);

exec crearpersona ('12345614', 'Isabel', 'Lopez', 'Expósito', '26/08/1939', '12345614', 'NordOest',
45, 6, '3-4', 17403, 'Sant Hilari', 677654345);

exec crearpersona ('12345615', 'Pilar', 'García', 'García', '21/09/1955', '12345615', 'Sudoest',
15, 5, '2-c', 17403, 'Sant Hilari', 972456665);

exec crearpersona ('12345616', 'Albert', 'Miró', 'Sanchez', '01/01/1960', '12345616', 'PolNord',
60, 1, '2-3', 17403, 'Sant Hilari', 972876632);

```

4.1.2. METGE

```

exec crearmetge ('12345610', '1', '1', '00000000');
exec crearmetge ('12345679', '2', '1', '00000000');

```

4.1.3. ATS

```

exec crearAts ('12345612', '1', '12345610');
exec crearAts ('12345613', '1', '12345679');

```

Ara ja podem modificar les dades a la taula METGE per assignar els ATS corresponents:

```

exec modificarmetge ('12345610', '1', '1', '12345612');
exec modificarmetge ('12345679', '2', '1', '12345613');

```


4.1.4. PACIENT

```
exec crearpacient ('12345621', '12345610');
exec crearpacient ('12345678', '12345610');
```

4.1.5. HORARILABORAL

```
exec modificahorarilaboral ('12345610', 1, 1);
exec modificahorarilaboral ('12345679', 1, 2);
exec modificahorarilaboral ('12345613', 1, 2);
exec modificahorarilaboral ('12345612', 1, 1);
```

4.2. CREARHORAVISITA

Paràmetres d'entrada: data / hora d'inici, data / hora fi, metge, pacient, observacions.

Procediments implicats: CREARHORAVISITA.

Funcions Implicades: NOHIES, TEMPSVISITACORRECTE, METGEDISPONIBLE.

Nº	Paràmetres	Resultat esperat	Valor RSP Obtingut (Taula Log)
1	09-01-2006 09:00 09-01-2006 09:20 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el temps de visita no compleix el paràmetre de temps de visita mínim (30')	(CREARHORAVISITA) ERROR AL INSERIR HORA A HORAVISITA, TEMPS VISITA FORA PARAMETRES ErrorMsg = User-Defined Exception ErrorNum = 1
2	10-01-2006 09:00 10-01-2006 09:30 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el metge no treballa el dimarts.	(METGEDISPONIBLE) ERROR A METGE DISPONIBLE, Traça = 5 ErrorMsg = ORA-01403: no se han encontrado datos ErrorNum = 100 (CREARHORAVISITA) ERROR A CREAR HORA VISITA, Traça = 5 ErrorMsg = ORA-06503: PL/SQL: La función no ha devuelto ningún valor ErrorNum = -6503
3	09-01-2006 16:00 09-01-2006 16:30 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el metge visita durant el torn de matí.	(METGEDISPONIBLE) ERROR AL COMPROVAR TORN DE TREBALL, TORN INCORRECTE ErrorMsg = User-Defined Exception ErrorNum = 1 (CREARHORAVISITA) ERROR A CREAR HORA VISITA, Traça = 5 ErrorMsg = ORA-06503: PL/SQL: La función no ha devuelto ningún valor ErrorNum = -6503
4	09-01-2006 09:00 09-01-2006 09:30 12345610 12345621 Pacient molt delicat. 1 12345678	Hora de visita inserida correctament	(CREARHORAVISITA) INSERTAR HORA VISITA O.K.
5	09-01-2006 09:15 09-01-2006 09:45 12345610 12345621 Pacient molt delicat. 1 12345678	Error, hora de visita ocupada	(CREARHORAVISITA) ERROR AL INSERIR HORA A HORAVISITA, HORA OCUPADA ErrorMsg = User-Defined Exception ErrorNum = 1

4.3. MODIFICARHORAVISITA

Paràmetres d'entrada: idHoravisita, data / hora d'inici, data / hora fi, metge, pacient, observacions.

Procediments implicats: MODIFICARHORAVISITA.

Funcions Implicades: NOHIES, TEMPSVISITACORRECTE, METGEDISPONIBLE.

Nº	Paràmetres	Resultat esperat	Valor RSP Obtingut (Taula Log)
1	1 09-01-2006 09:10 09-01-2006 09:30 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el temps de visita no compleix el paràmetre de temps de visita mínim (30')	(MODIFICARHORAVISITA) ERROR AL MODIFICAR HORA A HORAVISITA, TEMPS VISITA FORA PARAMETRES ErrorMessage = User-Defined Exception ErrorNum = 1
2	1 10-01-2006 09:30 10-01-2006 10:00 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el metge no treballa el dimarts.	(METGEDISPONIBLE) ERROR A METGE DISPONIBLE, Traça = 5 ErrorMessage = ORA-01403: no se han encontrado datos ErromNum = 100 (MODIFICARHORAVISITA) ERROR A MODIFICAR HORA VISITA, Traça = 9 ErrorMessage = ORA-06503: PL/SQL: La función no ha devuelto ningún valor ErrorNum = -6503
3	1 09-01-2006 16:30 09-01-2006 17:00 12345610 12345621 Pacient molt delicat. 1 12345678	Error, el metge visita durant el torn de matí.	(METGEDISPONIBLE) ERROR AL COMPROVAR TORN DE TREBALL, TORN INCORRECTE ErrorMessage = User-Defined Exception ErromNum = 1 (MODIFICARHORAVISITA) ERROR A MODIFICAR HORA VISITA, Traça = 9 ErrorMessage = ORA-06503: PL/SQL: La función no ha devuelto ningún valor ErrorNum = -6503
4	1 09-01-2006 09:15 09-01-2006 09:45 12345610 12345621 Pacient molt delicat. 1 12345678	Hora visita modificada correctament.	(MODIFICARHORAVISITA) MODIFICAR HORA VISITA O.K.
5	1 09-01-2006 09:15 09-01-2006 09:45 12345610 12345621 Pacient millor. 1 12345678	Hora visita modificada correctament.	(MODIFICARHORAVISITA) MODIFICAR HORA VISITA O.K.

4.4. ELIMINARPERSONA

Paràmetres d'entrada: idPersona

Procediments implicats: ELIMINARPERSONA.

Funcions Implicades: NOHIES, ELIMINARRELACIONS.

Nº	Paràmetres	Resultat esperat	Valor RSP Obtingut (Taula Log)
1	12345690	No existeix la persona	(ELIMINARPERSONA) ERROR A ELIMINAR PERSONA, NO EXISTEIX A LA TAULA PERSONA ErrorMsg = User-Defined Exception ErrorNum = 1
2	12345610	Persona eliminada de la taula PERSONA. Metge eliminat de la taula METGE. Metge ATS substituït per '0' a la taula ATS. Metge assignat substituït per '0' a la taula PACIENT. Horari laboral referent a aquest metge eliminat a la taula HORARILABORAL.	(ELIMINARPERSONA) ELIMINAR PERSONA O.K.

5. Grandària de la base de dades

Fem un càlcul aproximat de l'espai de memòria que ocuparà la base de dades, per fer-ho definirem l'abast del nostre centre mèdic, que estarà format per :

- 2000 Clients, possibles pacients.
- 10 Metges.
- 10 ATS.
- 5 Administratius.

Considerarem que totes les hores disponibles de visita estan ocupades, sabem que l'horari de treball està distribuït en torns, matí i tarda de sis hores cadascun, si el temps mínim per visita el considerem de trenta minuts, el CM realitzarà un total de dotze visites diàries per metge, també comptarem amb una mitjana de cinc urgències al dia.

Les dades de la fitxa de visita, es podran consultar directament fins a una antiguitat de quatre anys, a partir d'aquesta data, es disposaran en un arxiu històric, convenientment protegit i emmagatzemat en un suport que garanteixi la integritat de les dades, en el moment de la mort d'un pacient, les seves dades personals i les fitxes de visita es guardaran a l'històric fins a un període de tres anys, després seran eliminades. Les dades del personal que treballa al CM s'eliminaran quan no hi hagi relació laboral.

A partir d'aquestes especificacions, calculem la grandària de les dades taula per taula, primer calculem l'espai que ocupa un sol registre de la taula, tenint en compte que partirem d'ocupació màxima de camp. Agruparem els tipus de dades per obtenir la grandària.

- **Persona:**
 - VARCHAR: (118) * 1 byte = 118
 - DATE: (1) * 7 bytes = 7
 - INTEGER: (3) * 4 bytes = 12

Total: 137 bytes

- **Medicament:**
 - VARCHAR: (165) * 1 byte = 165
 - Total: 165 bytes

- **Especialitat:**
 - VARCHAR: (30) * 1 byte = 30
 - Total: 30 bytes

- **Estat:**
 - VARCHAR: (30) * 1 byte = 30
 - Total: 30 bytes

- **Acces:**
 - VARCHAR: (20) * 1 byte = 20
 - Total: 20 bytes

- **Torn:**
 - VARCHAR: (30) * 1 byte = 30
 - Total: 30 bytes

- **Diasetmana:**
 - VARCHAR: (20) * 1 byte = 20
 - Total: 20 bytes

- **Recepta:**
 - VARCHAR: (25) * 1 byte = 25
 - Total: 25 bytes

- **Pacient:**
 - VARCHAR: (20) * 1 byte = 20
 - Total: 20 bytes

- **Metge:**
 - VARCHAR: (40) * 1 byte = 40
 - Total: 40 bytes

- **ATS:**
 - VARCHAR: (30) * 1 byte = 30
 - Total: 30 bytes

- **Administratiu:**
 - VARCHAR: (20) * 1 byte = 20
 - Total: 20 bytes

- **Horavisita:**
 - VARCHAR: (60) * 1 byte = 60
 - DATE: (2) * 7 bytes = 14

Total: 74 bytes

- **Horarilaboral:**
 - VARCHAR: (30) * 1 byte = 30
 - INTEGER: (2) * 4 bytes = 8

Total: 38 bytes

- **Fitxavisita:**
 - VARCHAR: (650) * 1 byte = 650
 - DATE: (2) * 7 bytes = 14

Total: 664 bytes

- **Parametre:**
 - VARCHAR: (30) * 1 byte = 30

Total: 30 bytes

- **Registrelog:**
 - VARCHAR: (940) * 1 byte = 940
 - DATE: (1) * 7 bytes = 7

Total: 947 bytes

Sabent la grandària de cada registre, ara calcularem el que ocupa cada taula, hi haurà taules que no variaran o que ho faran molt poc al llarg del temps, al contrari que d'altres que arribaran a la plenitud màxima quan portin quatre anys d'activitat, unes les anomenarem taules de grandària estable i les altres de grandària dinàmica.

Taules de grandària estable i el seu volum:

Taula	Espai registre (bytes)	Total registres	Total espai ocupat (bytes)
Persona	137	2025	277425
Medicament	165	4000	660000
Especialitat	30	10	300
Estat	30	5	150
Accés	20	2	40
Torn	30	3	90
Diaetmana	20	7	140
Pacient	20	2000	40000
Metge	40	10	400
ATS	30	10	300
Administratiu	20	5	100
Horarilaboral	38	175	6650
Parametre	30	1	30
Total:			985625

Taules de grandària dinàmica i el seu volum:

Calcularem la seva grandària quan es porti quatre anys d'activitat, a partir d'aquí, els registres de la taula Horavisita, Fitxavisita i Recepta es desaran en un altre suport i formaran part de l'històric, els altres s'eliminaran.

Considerarem: Un total de cent vint fitxes de visita per dia, amb una mitjana de dos medicaments receptats per visita i tot això gestionat amb un total de mil moviments de mitjana diaris a la base de dades.

Cinquanta dues setmanes a l'any, per cinc dies hàbils (considerem molt poca l'activitat del cap de setmana) tenim un total de 260 dies per 4 anys d'emmagatzematge.

Taula	Espai registre (bytes)	Total registres	Total espai ocupat (bytes)
Recepta	25	249600	6240000
Horavisita	74	124800	9176000
Fitxavisita	664	124800	82867200
Registrelog	947	1040000	984880000
			Total: 1083163200

Necessitem un total de 1084148825 bytes \approx 1034 Mbytes \approx 1Gbyte d'espai necessari aproximat per a les dades.

6. Per acabar...

Els objectius que ens proposàvem a l'inici eren:

Realitzar el disseny d'un sistema informàtic capaç de dur *part* de la gestió d'un consultori mèdic, ens centrarem en la gestió de la relació del pacient amb el consultori, partint del concepte d'historial mèdic, amb totes les fitxes de visita que el componen i de tots els processos que comporten una visita mèdica, des de l'establiment de l'hora de visita fins la sortida.

Per assolir-ho hem creat un sistema de gestió de dades generades per un consultori mèdic, amb taules i procediments emmagatzemats necessaris per:

- Poder realitzar altes, baixes i modificacions de pacients.
- Gestionar l'agenda del personal mèdic per atendre els pacients, hores de visita.
- Guardar el diagnòstic, medicaments receptats o informació generada per cada visita mèdica.
- Gestió de medicaments amb la descripció, característiques...

Totes aquestes dades formen part d'un historial mèdic per a cada pacient, on es tindrà accés a totes les dades esmentades abans.

A més, els procediments emmagatzemats dissenyats, contenen com a mínim les següents característiques tècniques:

- Un paràmetre de sortida anomenat RSP, de tipus *string*, que indicarà si l'execució ha finalitzat amb èxit (valor 'OK') o si ha fracassat (valor 'ERROR + tipus d'error').
- Tractament d'excepcions.
- Registrar totes les crides a procediments que es facin en una taula de log, on contindrà informació de cada crida i els resultats obtinguts.

6.1. Comentaris de disseny

El disseny i Implementació de tot el sistema ha estat una tasca costosa, s'ha pres el model de funcionament d'un CAP, centre d'assistència primària, d'aquesta manera ens hem acostat el més possible a la realitat.

Hem aconseguit un sistema àgil, on amb poques taules i pocs procediments d'accés es poden dur a terme les tasques demanades pel centre mèdic. Només s'ha utilitzat tres tipus de dades diferents: Varchar, Integer i Date, que son fàcils de manejar i permeten guardar totes les dades necessàries. La quantitat de memòria requerida és poca, la base de dades pot funcionar sense necessitar cap extra de maquinari.

No s'utilitzen claus foranes, només claus principals, les relacions entre taules es controlen totes a través dels procediments, d'aquesta manera es té un major control sobre els excepcions, podent-les personalitzar i també més flexibilitat a la hora d'eliminar les dades.

6.2. Proves

Tots els procediments han sigut provats exhaustivament obtenint els resultats esperats, en aquesta memòria només es comenten les proves als procediments que es consideren més complexes, però el mateix protocol s'ha utilitzat pels altres. Cal dir que és molt gratificant veure que després de tot l'esforç de programació i correccions de codi, els procediments realitzen el que tu demanes.

6.3. Programari

Abans de la realització d'aquest projecte, Oracle era totalment desconegut, havia sentit a parlar d'ell amb molt bones referències, però ara he pogut comprovar l'enorme potencial que presenta, s'ha de tenir en compte en projectes futurs, a més no s'ha de deixar escapar tot l'aprenentatge obtingut en la realització d'aquest treball. Esmentar també l'ajuda del PS/SQL Developer, que ha facilitat la programació i proves del codi, gràcies al seu concepte d'entorn, ajuda a tenir una visió més clara d'Oracle.

6.4. Fonts d'informació

Per a la realització d'aquest projecte s'ha utilitzat la següent documentació:

- (O_reilly)-_Oracle_PL-Sql_Programming_3rd_Ed.pdf
- Les ajudes tant d'Oracle com del PL/SQL.
- El Google, on s'han consultat infinitat de fòrums, cursos, informes i ajudes en línia.