Implementació d'un esquema criptogràfic per gestionar de forma segura els historials mèdics dels pacients a través d'una xarxa de comunicacions

Jordi Piqueras Bautista
Enginyeria en Informàtica

Jordi Castellà Roca
Consultor

7 de gener de 2008

Resum

Des de fa ja alguns anys, l'ús d'Internet i, en general, de les xarxes de comunicació, ha experimentat un notable creixement. La constant evolució i l'expansió tecnològica, dins d'un marc que ofereix una extensa oferta de serveis i d'aplicacions d'origen divers i variat, han estat els factors determinants. Tot això, permet posar a l'abast tot un ventall d'avantatges i de possibilitats pel que fa a l'accés a fonts de dades remotes i per a l'intercanvi d'informació.

Si ens centrem dins de l'àmbit de la sanitat, escenari en el qual s'emmarca aquest projecte, es posa de manifest el valor afegit que es pot aconseguir gràcies als avantatges que se'n desprenen en l'ús i les possibilitats que ofereixen les xarxes tecnològiques. Permetre a un metge poder consultar l'historial d'un pacient de manera ràpida, en un moment donat i des d'una ubicació física diferent en la que pot estar emmagatzemada la informació, pot ajudar al metge a emetre un ràpid diagnòstic i preparar les actuacions que es considerin oportunes per a un possible tractament.

D'altra banda, hom coneix que les dades incloses en els historials mèdics dels pacients són de caire estrictament confidencial i només hi poden tenir accés determinat personal sanitari autoritzat, i, el propi pacient. Per aquest motiu, es fa necessari l'establiment de fortes mesures de seguretat per a preservar la integritat de la informació continguda en els historials i per a poder garantir la privacitat de les dades.

En aquest projecte final de carrera es presenta un sistema capaç de gestionar i emmagatzemar les històries mèdiques dels pacients. El sistema permetrà realitzar operacions de lectura i modificació de dades sobre els expedients mèdics de manera segura i fiable tenint en compte que els accessos a la informació s'efectuen a través d'una xarxa de comunicació.

Donades les característiques i propietats de les dades esmentades, cal adoptar mecanismes necessaris per a garantir el màxim nivell de seguretat. El sistema que es presenta proposa una solució que utilitza la criptografia asimètrica de clau pública i defineix tot un conjunt de protocols criptogràfics per a garantir totes aquestes necessitats de seguretat.

Índex

Índex de figures

1.Introducció

1.1. Justificació del Projecte Final de Carrera i context en el que es desenvolupa

Avui en dia, Internet i, en general, les xarxes de comunicació, es projecta com un entorn o mitjà que ofereix un conjunt de serveis i d'utilitats molt extens. Aquest fet, el qual ha estat fruit de la constant evolució tecnològica que proporcionen innovadors mecanismes per a l'intercanvi d'informació, possibilita el desenvolupament de noves concepcions de negoci i d'aplicacions d'origen divers i variat.

L'intercanvi d'informació o de coneixement de manera ràpida i eficaç es converteix en un factor clau per tal de que aquesta innovació tecnològica es pugui extendre per a donar suport a les necessitats emergents.

Aquestes xarxes, a més de possibilitar l'accés ràpid a grans volums d'informació, permeten també establir comunicacions per a l'intercanvi de dades en qualsevol moment donat, podent ésser efectuades des d'ubicacions llunyanes físicament. De manera addicional, la seguretat es converteix clarament en un altre requeriment, el qual es pot percebre com a determinant, tenint en compte la gran varietat d'escenaris aplicables.

La sanitat és un dels escenaris possibles en el que es poden percebre notablement els beneficis en l'ús de les xarxes de comunicació. Permetre a un metge poder consultar l'historial d'un pacient de manera quasi instantània, pot ajudar a emetre un diagnòstic escaient per al pacient i prendre una decisió adequada.

Un fet rellevant, d'altra banda, és que la informació existent en els historials mèdics és de caràcter estrictament confidencial, i, aquesta, només pot ésser accedida per persones autoritzades. És per això, que es fa necessari l'establiment de fortes mesures de protecció per tal de preservar la informació.

La justificació d'aquest projecte rau, fonamentalment, en poder disposar d'un sistema informàtic capaç de gestionar els historials mèdics dels pacients, i que utilitza una xarxa de comunicacions com a mitjà per a poder realitzar l'intercanvi de dades i/o l'accés a la informació. Tenint en compte

les propietats inherents a les xarxes, els possibles riscos o amenaces emergents, i el valor de la informació continguda en els historials mèdics, aquest sistema ha de seguir un esquema criptogràfic que garanteixi totes les necessitats de seguretat existents.

1.2. Objectius del projecte

L'objectiu principal d'aquest projecte, és implementar un sistema software que permeti gestionar historials mèdics de pacients a través d'una xarxa de comunicació, garantint que es compliran les necessitats de seguretat per a protegir la informació continguda en els historials mèdics dels pacients.

D'una banda, el sistema ha de permetre que el personal sanitari autoritzat pugui accedir a la consulta i a la modificació de les dades contingudes en els historials mèdics dels seus pacients de forma segura i fiable.

Tanmateix, es preveu la implementació d'una part del sistema encarregada d'integrar aquelles funcionalitats per a fer possible que els pacients puguin consultar les dades del seu propi historial mèdic.

Un contenidor ha de ser l'encarregat d'emmagatzemar el conjunt dels historials mèdics dels pacients, i, aquest, ha d'englobar, de forma addicional, mecanismes per a fer possible dur a terme una òptima gestió. Aquest contenidor, o gestor central, també s'encarrega d'incorporar funcions que permeten donar d'alta al sistema els pacients i els metges, i d'identificar i de validar als usuaris que hi accedeixen.

Donat el gran valor de la informació continguda en els historials mèdics, és, doncs, un dels punts més rellevants d'aquest projecte, la confecció d'un esquema criptogràfic capaç de garantir les necessitats de seguretat inherents del sistema que es desenvolupa. D'aquesta manera, s'assegura que es compleixi que, la confidencialitat, l'autenticitat, la integritat i la validesa de la informació continguda en els historials mèdics, esdevingui present en tot moment, i, que les accions efectuades per qualsevol persona, ja sigui per consultar, modificar o inserir informació, no han de poder-se posar en dubte.

Mitjançant l'ús de la criptografia asimètrica de clau pública i la implementació de diversos protocols criptogràfics, es durà a terme la confecció de l'esquema criptogràfic per a garantir les propietats de seguretat que el tractament de la informació requereix.

Per tal de fer accessibles remotament els historials mèdics dels pacients, s'utilitzarà el protocol de comunicació *Remote Method Invocation*. Amb aquesta tecnologia s'estableix la base per a instal·lar i configurar un servidor que permeti realitzar i invocar crides de forma remota que permetin dur a terme els accessos i les operacions.

En el projecte s'utilitzarà el llenguatge de marques XML per a fer l'intercanvi de dades entre el client i l'aplicació durant l'execució dels protocols criptogràfics implementats.

Un sistema gestor de base de dades serà l'encarregat d'enregistrar i emmagatzemar les dades que intervenen en la configuració dels historials mèdics dels pacients.

Finalment, cal dur a terme el disseny i la confecció de la interfície gràfica, a través de la qual els diferents usuaris de l'aplicació interactuaran per a poder realitzar les operacions. Es fa necessari l'existència d'una interfície gràfica adaptada a cadascun dels perfils d'usuari de l'aplicació.

1.3. Enfocament i mètode seguit

D'acord amb els objectius plantejats, es poden identificar les diferents fases en les que es pot estructurar el projecte. Alhora, s'estableix un procediment de treball gradual i progressiu. Des d'aquesta perspectiva, dividint el projecte en diferents parts i fent ús d'una implementació incremental, es pot seguir una filosofia de test unitari per a cadascuna de les fases, amb la finalitat de poder integrar cada fase següent amb la seva anterior.

Així doncs, el projecte consta de les següents parts:

- IAIK i PKI: la instal·lació de les llibreries criptogràfiques IAIK i l'ús de la tecnologia PKI (infraestructura de clau pública), permet als usuaris autenticar-se enfront al sistema i utilitzar la informació dels certificats per xifrar i desxifrar missatges, xifrar digitalment la informació i garantir la propietat de no repudi, entre altres.
- Esquema criptogràfic: és un dels punts clau d'aquest projecte sobre el qual es desenvolupa el sistema. Aquest esquema defineix tot el conjunt de protocols criptogràfics per a dur a terme qualsevol acció o servei.
- XML (*eXtensible Markup Language*): llenguatge amb el qual s'indica el format per a representar i transferir la informació que s'envia durant l'execució dels protocols criptogràfics.
- RMI (*Remote Method Invocation*): esdevé un mecanisme essencial per tal de poder realitzar crides a les funcionalitats del sistema de manera remota. És a dir, permet la comunicació dels diferents components mitjançant mètodes d'invocació remota.
- Base de dades: el seu ús ens permet emmagatzemar la informació continguda en els historials mèdics dels pacients.
- Interfícies gràfiques: la creació d'interfícies gràfiques permet dur a terme més fàcilment l'execució de les funcionalitats de les aplicacions del sistema. Hi ha d'haver una aplicació per a cada component: metge, pacient i gestor.
- Joc de proves: a mesura que es van desenvolupant les diferents etapes del projecte, cadascuna d'aquestes es va testejant i es va recopilant de manera individual.
- Documentació: revisió dels apartats i continguts de la memòria, lliurament i presentació del projecte.

1.4. Planificació del projecte

Aquest projecte es comprèn entre els dies 19 de setembre de 2007 i 7 de gener de 2008 corresponents al quadrimestre del curs de tardor de l'any 2007-2008. La periodificació de les diferents parts que composen el projecte és la que es detalla tot seguit.

Del 19 al 23 de setembre:

- Preparació de l'entorn de desenvolupament, instal·lació de les llibreries d'encriptació IAIK i creació dels certificats seguint la tecnologia d'infraestructura de clau pública (PKI). Descripció dels conceptes PKI.

Del 24 de setembre al 21 d'octubre:

- Implementació dels protocols: disseny de l'esquema criptogràfic i implementació dels protocols utilitzats. Proves de test i documentació dels conceptes criptogràfics.

Del 22 d'octubre al 4 de novembre:

- Representació de les dades: disseny i implementació per a representar les dades en format XML disseny, proves de test i documentació dels conceptes XML.

Del 5 de novembre al 18 de novembre:

- Comunicacions RMI: disseny i implementació de la comunicació entre els diferents components. Desenvolupament del servidor RMI i del client RMI. Proves i documentació dels conceptes RMI.

Del 19 de novembre al 2 de desembre:

- Base de dades: disseny del model conceptual i físic de la base de dades per l'emmagatzemament dels registres. Implementació, proves de test i documentació.

Del 3 de desembre al 18 de desembre:

- Vista client: disseny i implementació de la interfície del pacient. Proves i documentació.

Del 17 de desembre al 30 de desembre:

- Vista gestor: disseny i implementació de la interfície del gestor del sistema. Proves i documentació.

Del 31 de desembre al 6 de gener:

- Documentació: descripció de les conclusions i revisió de les diferents seccions de la documentació.

1.5. Productes obtinguts

Tal i com s'ha comentat anteriorment, l'objectiu d'aquest projecte recau en desenvolupar un sistema capaç de gestionar els historials mèdics. Mitjançant la implementació d'un esquema criptogràfic, el sistema ha de permetre l'accés a consulta i modificació dels historials mèdics, tenint en compte l'existència de diferents perfils d'usuaris.

Com a resultat de la implementació, s'obtindrà un sistema amb els components de programari següents:

Aplicació metge

Aquest mòdul permet que qualsevol metge autoritzat pugui accedir a la consulta i modificació dels historials mèdics existents de forma segura i fiable.

Aplicació pacient

La implementació d'aquest mòdul integra totes aquelles funcionalitats per a fer possible que un determinat pacient pugui consultar les dades del seu propi historial mèdic tenint en compte el rigor de seguretat establert.

Aplicació central

D'una banda aquesta abstracció engloba el conjunt d'historials mèdics dels pacients i la seva gestió, i, de l'altra, permet donar d'alta al sistema les persones a les quals se'ls hi concedeix els permisos de consulta i modificació d'historials.

1.6. Breu descripció dels capítols següents

Seguidament, es presenta una petita introducció dels continguts presents en aquesta documentació.

Estudi de les necessitats del sistema

Per a obtenir una visió concreta i ben definida de les necessitats que el sistema ha de cobrir, és necessari realitzar una anàlisi sobre els diferents actors que interaccionen amb el sistema, estudiar les funcionalitats, les accions i els serveis a proveir.

Infraestructura de clau pública

En aquest capítol s'introdueixen els conceptes generals de criptografia de clau pública, i s'enumeren i es descriuen els components principals per a fer ús de la infraestructura de clau pública.

Disseny de l'esquema criptogràfic

Una de les parts més importants d'aquest projecte són els esquemes criptogràfics. Aquests descriuen els protocols necessaris per a poder dur a terme les accions o serveis que ofereix el propi sistema.

Arquitectura del sistema

En aquest capítol es defineix l'arquitectura i l'estructura que forma l'entorn en el qual el sistema s'ha d'establir i que esdevindrà una guia per a poder concebre de manera adequada la implementació i el desenvolupament de les diferents parts que el composen.

Representació de les dades: XML

Es descriuen els conceptes de XML i s'explica el format de dades que se segueix per tal d'aconseguir que els clients i el gestor de l'aplicació s'entenguin en la seva comunicació d'intercanvi de dades.

Comunicació dels components: RMI

La comunicació dels diferents components és una part essencial d'aquest projecte. Per a aconseguir-ho, aquesta comunicació es realitza mitjançant la invocació de crides remotes. En aquest apartat s'expliquen els conceptes RMI més rellevants i com aquests s'apliquen al disseny del sistema de gestió d'historials mèdics.

Base de dades

L'emmagatzemament de les dades i dels historials mèdics en una base de dades té per objectiu aconseguir disposar d'una bona estructura de la informació tractada. En aquesta secció es dóna el disseny de la base de dades i es descriuen les decisions preses.

Interfícies gràfiques: interfície del pacient, interfície del metge i interfície del gestor del sistema

Les interfícies gràfiques representen els components amb el que els clients del sistema interaccionen per a executar les funcionalitats i dur a terme les accions per a les quals el sistema és concebut. En aquest punt es parla com s'han confeccionat les interfícies del pacient, del metge i del gestor del sistema, en termes de disseny.

Joc de proves

En aquest apartat es detallen breument els passos a seguir per a dur a terme un cert conjunt de casos de prova.

Conclusions

En aquest apartat es contrasten els fets sorgits més destacats per assolir els objectius d'aquest projecte un cop elaborat el projecte.

Glossari

Capítol on es troben la definició dels termes i de les paraules clau utilitzades en aquesta documentació.

Bibliografia

Detall de les fonts d'informació en format paper i digital a les que s'han accedit per a la confecció d'aquest projecte.

Annexos

Relació d'un conjunt d'apartats amb informació addicional envers al projecte. Es descriuen la relació d'arxius i com preparar el sistema.

2. Estudi de les necessitats del sistema

2.1. Introducció

Per a obtenir una visió concreta i ben definida de les necessitats que el sistema ha de cobrir, és necessari realitzar una anàlisi sobre els diferents actors que interaccionen amb el sistema, estudiar les funcionalitats, les accions i els serveis a proveir. D'altra banda, també es fa necessari determinar com es gestionarà la informació que envolta el sistema. Finalment, es destacaran els requeriments pel que fa a seguretat que s'han de garantir i que formen part dels objectius principals d'aquest projecte.

2.2. Actors

El sistema és concebut per a respondre a les necessitats d'ús d'un petit grup de perfils d'usuaris. Tenint en compte el context en què es desenvolupa aquest projecte, emmarcat dins del sector de la sanitat, el sistema bàsicament serà utilitzat per part dels metges amb accés a les dades d'historials mèdics, i per part dels pacients o persones amb història mèdica que han de tenir accés al seu propi historial clínic.

D'altra banda, es pot considerar la figura de l'administrador del sistema com a tercer perfil existent. L'administrador s'encarrega de donar d'alta als usuaris (pacients i metges), a més de fer l'assignació en el sistema dels pacients que té cada metge, i, assegurar la posada en marxa del sistema.

Per tant, els actors que es reconeixen en l'ús del sistema, són els següents:

- Pacients: aquest rol es refereix a aquells usuaris que accediran al seu historial mèdic per a només consulta. En cap cas aquest actor pot modificar les dades contingudes en la seva història clínica.

- Metges: aquest perfil es refereix al conjunt d'usuaris que formen part del personal sanitari amb accés als historials mèdics dels pacients i que poden realitzar tasques de consulta i modificació de les dades de la història clínica dels pacients.

- Gestor del sistema: és l'actor que representa a l'administrador del sistema. El gestor del sistema s'encarrega de donar d'alta als pacients i al personal mèdic al sistema de manera manual. També serà l'encarregat de la posada en marxa del sistema.

2.3. Accions i serveis del sistema

2.3.1. Introducció

Un cop identificats els actors que interactuen amb el sistema, el següent pas consisteix en analitzar i descriure les funcionalitats que cal desplegar sobre aquest i que seran executades pels propis actors.

2.3.2. Gestió dels usuaris

La gestió dels usuaris es farà a càrrec de l'administrador del sistema. En principi, és el propi administrador l'encarregat de donar d'alta o baixa a un usuari determinat. Addicionalment, vetllarà i aplicarà els esforços i accions necessàries per a que el sistema estigui en operació.

2.3.3. Autenticació dels usuaris

Tots els usuaris, tant els pacients com els metges, hauran d'autenticar-se a l'aplicació per tal de poder-ne fer ús. En el procés d'autenticació, es verificarà l'autenticitat de l'usuari mitjançant els mecanismes de criptografia de clau pública, validant de manera correcta, segura i fiable, la identitat de l'usuari que inicia l'aplicació.

2.3.4. Consulta d'un expedient mèdic

Cada actor del sistema pot consultar una història mèdica. En funció del rol i del tipus d'actor, es tindrà accés a la informació continguda a l'expedient en qüestió que es vol consultar. Així, per exemple, els pacients només tindran accés de consulta per al seu propi historial. D'altra banda, els metges podran consultar l'historial de qualsevol pacient, excepte en el cas que existeixi informació en l'expedient considerada de caràcter privat i que només hi tindrà accés el metge assignat al pacient en qüestió.

2.3.5. Consulta dels pacients assignats a un metge

Els metges han de tenir accés al sistema per a poder veure quins són els seus pacients assignats. Cada metge, tindrà un determinat nombre de pacients assignats i dels quals es farà càrrec.

2.3.6. Inserció de dades a l'expedient

Els metges podran incorporar dades a l'historial mèdic d'un pacient, sempre que aquest sigui un dels seus pacients assignats.

2.3.7. Finalitzar Sessió

Els usuaris de l'aplicació han de poder abandonar el sistema de manera segura després d'haver passat prèviament pel procés d'autenticació.

2.4. Diagrama de casos d'ús

Per a determinar els requeriments principals del sistema a confeccionar s'ha seguit la metodologia anomenada *UML* (*Unified Modeling Language*) i poder abstraure les accions i les funcions principals i que el sistema ha de complir.

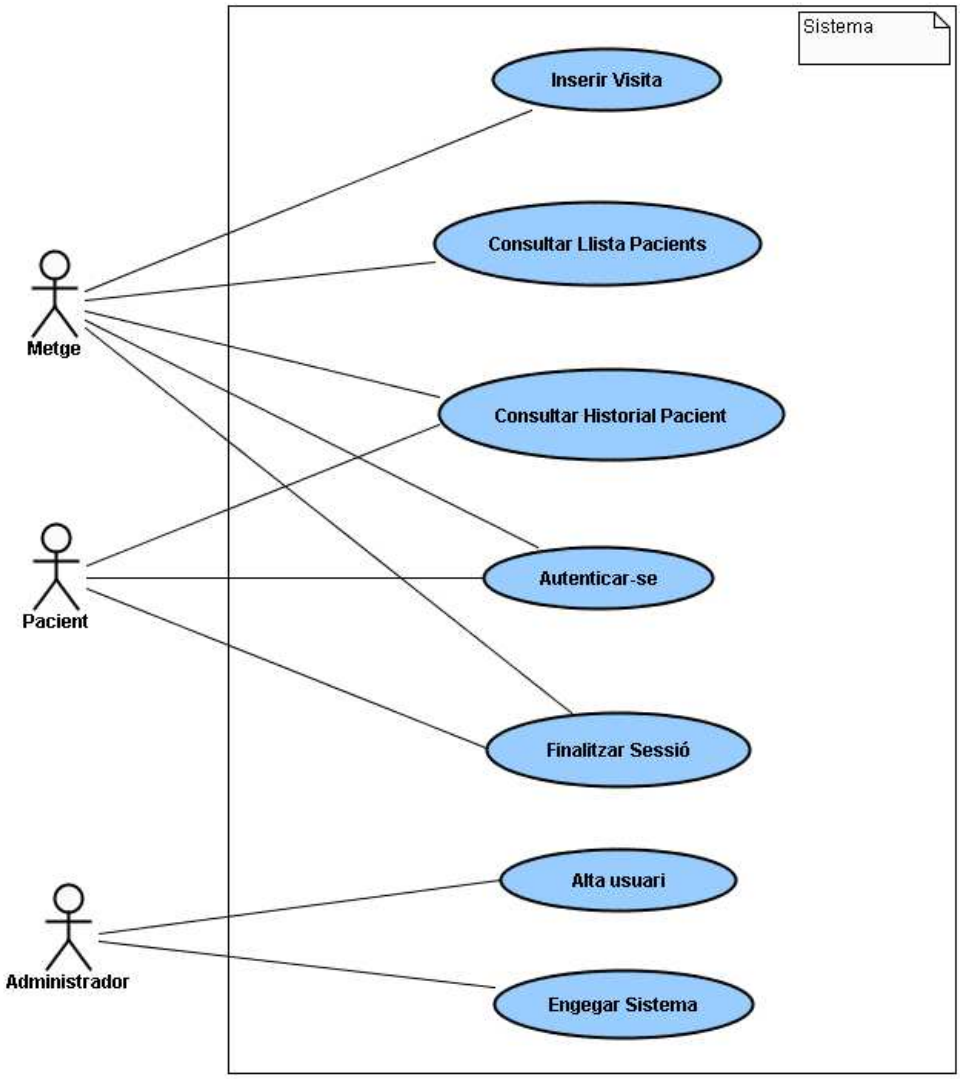

Figura 2-1. Diagrama de casos d'ús

El diagrama de casos d'ús mostra de manera conceptual les accions i funcions que el sistema recull i que s'han detectat durant l'estudi del sistema. Mostra un conjunt de casos d'ús, actors i les seves relacions.

Com es pot observar en la Figura 2-1, el sistema serà utilitzat principalment per metges i pacients els quals duran a terme les accions que li són permeses.

Es pot dir que l'administrador o gestor del sistema representa una figura secundària donat que tan sols s'encarregarà de donar d'alta els usuaris mitjançant un procés manual, i d'assegurar-se la correcta execució (o posada en marxa) del sistema. Per aquest motiu, és considera que no cal que s'autentiqui.

2.5. Gestió de la informació

2.5.1. Introducció

La principal informació que tractarà i gestionarà el sistema gira al voltant de les dades del pacient i configuraran el seu propi expedient. L'expedient del pacient inclourà tota aquella informació referent a les dades personals i de contacte del pacient, les dades mèdiques que seran públiques i visibles també per a qualsevol metge, i l'historial de visites. L'historial de visites és el contingut que engloba totes les visites que s'han efectuat per un metge a un determinat pacient, i que conformen la història dels diagnòstics, tractaments i l'evolució de les malalties i/o enfermetats detectades del pacient.

Configuració de la informació continguda en un expedient d'un pacient:

- Dades personals: NIF, nom, cognoms, adreça, població, telèfon i data de naixement.
- Dades mèdiques: pes, alçada i grup sanguini.
- Historial de visites: data de visita o d'ingrés del pacient, diagnòstic, tractament, evolució i marca de la confidencialitat de la visita (privada o no).

2.5.2. Classificació segons la privadesa

L'historial mèdic d'un pacient pot contenir informació molt diversa. Aquesta informació pot ser considerada més confidencial que una altra en funció del diagnòstic identificat. Aquest fet determina la possibilitat que existeixin malalties o enfermetats detectades que siguin de caràcter privat i que, per tant, només puguin ser visibles per al propi metge que duu el cas o l'historial del pacient.

D'aquesta manera, existeix la possibilitat de que un metge pugui indicar la particularitat de que una determinada visita efectuada per a un pacient, esdevingui de caràcter privat o confidencial. Tanmateix, aquesta visita, només serà visible o pública per al propi metge que s'encarrega de portar al pacient en qüestió.

2.5.3. Flux d'informació

El flux d'informació determina la relació dels permisos existents entre els actors i les dades o les diferents parts d'un expedient.

En primer terme, i per facilitar la gestió de la inserció de les dades dels pacients, es considera que el gestor del sistema serà l'encarregat de donar d'alta els pacients i d'incorporar la informació referent a les dades personals i mèdiques (alçada, pes i grup sanguini) que configuren l'expedient dels pacients. La figura del gestor es pot veure com a una persona encarregada d'administrar aquestes dades en el moment d'alta dels pacients, al mateix temps que s'encarrega de generar un certificat d'acord amb les necessitats establertes pel funcionament correcte del sistema.

D'altra banda, els metges seran els encarregats d'incloure la informació referent a la història mèdica dels pacients. Malgrat aquest fet, cal tenir en compte que només el propi metge assignat al pacient serà qui pugui realitzar la inserció d'aquest tipus de dades i incloure-les a l'historial de visites. D'aquesta manera, es considera que tot pacient tindrà un únic metge assignat que tingui accés a la inserció de la informació d'una visita determinada.

Finalment, tant els metges com els pacients, se'ls hi permetrà tenir accés de consulta sobre les dades personals, les dades mèdiques i la informació continguda a l'historial de visites. Cal remarcar, però, que els pacients només tindran accés de consulta al seu propi expedient, i que els metges podran consultar informació d'expedients d'altres pacients, sempre tenint en compte el caràcter confidencial que se li pugui donar a determinades visites. En aquest darrer cas, tan sols podrà disposar d'accés de consulta a les visites identificades com a privades el metge assignat del pacient.

A continuació, es presenta una taula en la que es representa aquest flux d'una manera esquematitzada.

Informació	Gestor	Metge	Pacient
Dades personals	Lectura, inserció i modificació.	Lectura	Lectura
Dades mèdiques	Lectura, inserció i modificació.	Lectura	Lectura
Historial mèdic		Lectura i inserció	Lectura

Figura 2-2. Flux d'informació

2.6. Requisits de seguretat

Al punt anterior s'ha classificat la informació i el fluxe de la mateixa, de manera que se sap qui hi té accés respecte els permisos de cada actor en relació al contingut de dades. En aquest sentit, cal enumerar les propietats de seguretat que cal complir per a poder solucionar les amenaces que envolten al sistema i que és necessari cobrir de manera precisa per a cadascun dels servei que es poden posar en execució.

- Privacitat de la informació: s'ha de preservar la confidencialitat de les dades de l'historial mèdic dels pacients. Es requereix i es fa estrictament necessari que la informació només pugui ésser visible únicament per a aquelles entitats o aquells actors que hi tenen permís.

- Autenticitat de continguts i d'actors: la informació que es guarda en el sistema ha de disposar d'una prova de la seva autenticitat. És a dir, cal disposar d'algun tipus d'identificació que atribueixi la correctesa de l'origen de la informació i que l'entitat o l'actor emissor no és un impostor.

- Protecció de la integritat de la informació: un cop la informació ha estat generada, s'ha de garantir en tot moment la seva integritat. Tan sols s'ha de permetre la manipulació de la informació per als actors identificats i autoritzats.

- Autoria d'una acció: si un usuari del sistema fa una certa acció, més tard no pot negar haver-la realitzat. Aquesta propietat és aplicable sobretot al registre de les visites que es realitzin sobre els expedients mèdics dels pacients.

En el proper capítol s'introdueixen els conceptes generals de criptografia de clau pública, i s'enumeren i es descriuen els components principals per a fer ús de la infraestructura de clau pública, necessària per a satisfer els objectius de seguretat que el sistema ha de complir.

3. Infraestructura de clau pública

3.1. Introducció

Tal i com s'ha plantejat anteriorment, un dels objectius principals d'aquest projecte és aconseguir establir un marc segur i fiable a l'hora de produir l'intercanvi d'informació entre els diferents participants o usuaris. Dit en altres paraules, donades les característiques del medi per on transcorre la informació, aquesta es pot veure afectada per l'existència d'amenaces. És per això, que cal cercar mecanismes i funcions que garanteixin uns procediments de seguretat per a poder preservar i protegir la informació de qualsevol possible amenaça a la qual pugui estar exposada.

Principalment, s'identifiquen quatre tipus de propietats que cal complir per a poder mantenir els criteris de seguretat:

- Confidencialitat: cal evitar qualsevol tipus de filtratge que es pogués donar per a tot individu que no sigui el propi emissor o el propi destinatari en una determinada transmissió. S'ha de garantir la privadesa de la informació.

- Integritat: s'ha de dissuadir a tot individu amb intenció de modificar i manipular la informació que és transmesa des d'un emissor a un determinat destinatari. El contingut de la informació ha de ser el mateix durant el procés de comunicació. És a dir, cal garantir que el destinatari rep el mateix missatge que l'emissor envia originalment.

- Autenticitat: és necessari poder garantir l'autoritat de les dades no es pugui veure afectada en cap moment. No ha d'existir la possibilitat de modificar l'autoria d'un determinat contingut.

- No-repudi: qualsevol participant que hagi promogut l'intercanvi d'informació o ha efectuat algun tipus d'acció sobre les dades, aquest, no se'n pot desdir de la seva autoria. Cap individu pot revocar ésser l'autor si ha estat ell mateix l'originador o el creador de les dades.

Per aconseguir evitar aquestes amenaces, s'utilitzen mètodes i procediments de criptografia de clau pública, o també anomenada, criptografia asimètrica.

Propòsit i funcionalitats

Es pot anomenar infraestructura de clau pública (*Public Key Infraestructure* corresponent a les sigles PKI en anglès) al conjunt de maquinari i programari, polítiques i procediments de seguretat, necessaris per a permetre assegurar la identitat dels participants en un intercanvi de dades utilitzant la criptografia. Tanmateix, la PKI permet l'execució amb garanties d'operacions criptogràfiques com el xifratge i desxifratge de la informació, la signatura digital o el no-repudi en comunicacions electròniques.

Com s'ha comentat, per a aquest projecte es fa necessari l'establiment d'un entorn en què la seguretat i la fiabilitat en l'intercanvi de dades esdevinguin en unes característiques integrades. Per aconseguir-ho, s'utilitza la criptografia de clau pública de manera que els usuaris i el gestor del sistema disposen cadascun d'una parella de claus diferents (una clau pública i una altra privada) i del seu corresponent certificat digital.

El certificat digital és una estructura de dades que conté informació del propietari del parell de claus criptogràfiques i la seva clau pública. Addicionalment, a aquests dos elements se'ls hi incorpora la signatura que els hi dóna validesa enfront a qualsevol usuari. Aquesta signatura, efectuada per un usuari o per una entitat externa lleial, assegura la integritat de la clau pública i del certificat i serveix com a mecanisme de protecció per evitar l'alteració o la modificació d'aquests.

D'altra banda, les claus han de ser prou fortes per minimitzar el risc de la probabilitat que un atacant les pugui trencar. Aquest fet depèn del conjunt de la longitud de la clau i de la qualitat de l'algorisme de generació de les claus.

Tanmateix, les operacions criptogràfiques de clau pública, són processos en els que s'utilitzen uns algoritmes de xifrat que són coneguts i són accessibles per a tothom. Per aquest motiu, la seguretat que pot aportar aquesta tecnologia, està fortament lligada a la privacitat de l'anomenada clau privada i dels procediments operacionals o polítiques de seguretat aplicats.

El propòsit d'una infraestructura de clau pública és aconseguir una gestió eficient i de confiança de les claus criptogràfiques i dels certificats per l'assoliment i l'acompliment de les propietats d'autenticació, integritat, no-repudi i confidencialitat, evitant d'aquesta manera, les amenaces a les que tot sistema o aplicació està exposat durant la comunicació.

Components principals

Els components més essencials que formen part d'una PKI són l'autoritat de certificació (notada per CA i corresponent al terme en anglès *Certification Authority*), els subscriptors i els repositoris entre d'altres. A continuació, es fa un petit incís aquests components i d'altres que sovint formen part dins d'una estructura típica PKI.

- Autoritat de certificació: és l'entitat de confiança que dóna validesa enfront a la relació d'una clau pública amb la identitat del seu titular. És també la responsable d'emetre i revocar certificats.

- Autoritat de registre: en anglès *Registration Authority*, l'RA és l'encarregada de verificar el lligam entre les claus públiques i la identitat dels seus titulars.

- Subscriptors i entitats finals: són aquells que posseeixen un parell de claus (pública i privada) i un certificat associat a la seva clau pública. Fan ús de la tecnologia PKI per a validar firmes digitals, xifrar i desxifrar documents, etc. Una entitat final representa un organisme, mentre que un subscriptor es refereix a una persona.

- Usuaris: són agents que validen signatures digitals i la seva ruta de certificació a partir de les claus emeses per autoritats de certificació de confiança. Poden xifrar documents i missatges per a subscriptors i entitats finals.

- Repositoris: són estructures encarregades d'emmagatzemar la informació relativa a la infraestructura de clau pública. Típicament, existeixen dos tipus de repositoris: un, on es guarden els certificats, i, un altre, on es guarden certificats revocats o considerats invàlids per algun motiu.

- Autoritat de validació: *Validation Authority*, la VA és l'encarregada de comprovar la validesa dels certificats digitals. Aquesta autoritat pot ser la pròpia autoritat de certificació o una entitat externa.

3.2. Generació de certificats

Introducció

En aquest apartat es descriu com s'han generat els certificats necessaris per a aquest projecte, seguint els passos necessaris i les comandes corresponents per a la construcció de la PKI.

En un primer terme es detalla el procediment per a poder crear una parella de claus i un certificat signat a partir d'aquestes dues claus. L'objectiu de crear aquest certificat autosignat és que passi a ser el certificat representant de l'autoritat certificadora CA.

A continuació, es descriuen els passos a seguir per crear una parella de claus, la corresponent petició de certificat a partir d'aquestes i l'emissió final del certificat basat en la petició de certificat i la pròpia CA. Fent ús d'aquests passos s'accedeix a poder crear la corresponent parella de claus i el certificat identificatiu per a cadascun dels clients que interaccionen amb el sistema: gestor, metge i pacient.

Finalment, es detalla com encapsular una parella de claus, el certificat corresponent, i, el certificat de l'autoritat de certificació, en un arxiu PKCS#12 (*Personal Information Exchange Syntax Standard*).

La creació dels certificats s'ha dut a terme mitjançant l'eina anomenada Openssl. Tanmateix, per comoditat i facilitat d'execució, s'ha utilitzat la mateixa contrasenya per a totes les claus i certificats: "uoc0506". A més, per determinar els paràmetres i camps per defecte del certificat, s'ha utilitzat el fitxer "openssl.conf".

Certificat de l'Autoritat Certificadora

1. Crear la parella de claus per l'Autoritat certificadora:

```
openssl genresa –des3 –rand aleatori –out CA.Key 2048
```

   La longitud de la parella de claus és de 2048 bits, xifrada amb Triple DES.

2. Generar un certificat autosignat amb la parella de claus de la CA (arxiu CA.key)

```
openssl req –new –sha1 –x509 –key CA.key –out CA.crt –days 365
```

S'utiltza la funció hash sha1 i obtenim el fitxer del certificat amb nom CA.crt.

Certificat Gestor

1. Generar una parella de claus per a l'usuari Gestor

```
openssl genrsa -des3 -out Gestor.key 1024
```

La longitud del parell de claus és de 1024 bits, i són xifrades amb Triple DES.

2. Generar la petició de certificat per a l'usuari Gestor

```
openssl req -new -sha1 -key Gestor.key -out Gestor.csr -config
openssl.conf
```

Funció resum (hash) sha1. El fitxer de sortida és Gestor.csr.

3. Generar el certificat per l'usuari Gestor

```
openssl x509 -req -in Gestor.csr -days 180 -CA CA.crt -CAkey CA.k
ey -CAcreateserial -extfile openssl.conf -extensions usr_cert -out
Gestor.crt
```

El certificat serà vàlid uns 180 dies. El fitxer amb el certificat és Gestor.crt.

4. Generar l'arxiu que conté la clau privada i el certificat en format PKCS#12

```
openssl pkcs12 -in Gestor.crt -inkey Gestor.key -name Gestor -cha
in -CAfile CA.crt -export -out Gestor.p12
```

El fitxer de sortida és: Gestor.p12

Certificat Pacient

1. Generar una parella de claus per a l'usuari Pacient

```
openssl genrsa -des3 -out Pacient.key 1024
```

La longitud del parell de claus és de 1024 bits, i són xifrades amb Triple DES.

2. Generar la petició de certificat per a l'usuari Pacient

```
openssl req -new -sha1 -key Pacient.key -out Pacient.csr –config
openssl.conf
```

Funció resum (hash) sha1. El fitxer de sortida és Pacient.csr.

3. Generar el certificat per l'usuari Pacient

```
openssl x509 -req -in Pacient.csr -days 180 -CA CA.crt -CAkey CA.
key -CAcreateserial -extfile openssl.conf -extensions usr_cert -out
Pacient.crt
```

El certificat serà vàlid uns 180 dies. El fitxer amb el certificat és Pacient.crt.

4. Generar l'arxiu que conté la clau privada i el certificat en format PKCS#12

```
openssl pkcs12 -in Pacient.crt -inkey Pacient.key -name Pacient -chain -
CAfile CA.crt -export -out Pacient.p12
```

El fitxer de sortida és: Pacient.p12

Certificat Metge

1. Generar una parella de claus per a l'usuari Metge

```
openssl genrsa -des3 -out Metge.key 1024
```

La longitud del parell de claus és de 1024 bits, i són xifrades amb Triple DES.

2. Generar la petició de certificat per a l'usuari Gestor

```
openssl req -new -sha1 -key Metge.key -out Metge.csr -config openssl.conf
```

Funció resum (hash) sha1. El fitxer de sortida és Metge.csr.

3. Generar el certificat per l'usuari Metge

```
openssl x509 -req -in Metge.csr -days 180 -CA CA.crt -CAkey CA.ke
y -CAcreateserial -extfile openssl.conf -extensions usr_cert -out
Metge.crt
```

El certificat serà vàlid uns 180 dies. El fitxer amb el certificat és Metge.crt.

4. Generar l'arxiu que conté la clau privada i el certificat en format PKCS#12

```
openssl pkcs12 -in Metge.crt -inkey Metge.key -name Metge -chain
-CAfile CA.crt -export -out Metge.p12
```

El fitxer de sortida és: Metge.p12

4. Disseny de l'esquema criptogràfic

4.1. Introducció

A partir de les anàlisi i estudis realitzats prèviament en els anteriors capítols, s'ha dissenyat un protocol criptogràfic per a cada acció que realitzen els usuaris, tant si es tracta d'un pacient com si es tracta d'un metge. Aquests protocols tenen com a objectiu garantir que el sistema compleixi en tot moment les propietats de seguretat identificades.

4.2. Notació emprada

A la descripció dels protocols criptogràfics que veurem a continuació, s'empra la següent notació:

- *K*: clau d'un criptosistema simètric.
- *M*: missatge.
- $E_K(M)$: xifratge simètric d'un missatge *M* amb la clau *K*.
- $D_K(C)$: desxifratge simètric del criptograma *C* amb la clau *K*.
- ($P_{Entitat}$, $S_{Entitat}$): parella de claus asimètriques propietat d'*Entitat*, on *P* correspon a la clau pública, i S a la privada.
- $S_{Entitat}\ [M]$: signatura digital del missatge *M* amb la clau asimètrica privada $S_{Entitat}$ d'*Entitat*.
- $P_{Entitat}\ [M]$: xifratge del missatge *M* amb la clau asimètrica pública $P_{Entitat}$ d'Entitat.
- *H(M)*: sortida d'una funció resum criptogràfica del missatge *M* (altrament anomenades funcions *hash*).
- N_i: valor aleatori.

- $Id_{Entitat}$: identificador d'*Entitat* dins del sistema.

4.3. Protocols

En aquest apartat es descriu l'esquema criptogràfic desenvolupat per a cada tipus d'acció o servei del nostre sistema. Cada acció o servei, es realitza d'acord a un protocol criptogràfic. Aquest conjunt de protocols sobre els quals es basen les funcionalitats del sistema, permeten garantir les necessitats de seguretat que envolten a les comunicacions realitzades a través de la xarxa de telecomunicacions.

4.3.1. Identificació

El sistema gestor ha de permetre discriminar les accions que pot dur a terme un usuari determinat en funció del seu rol associat. Aquest perfil o rol associat identificarà, doncs, el perfil de l'usuari en qüestió i determinarà les accions que se li permeten realitzar. Així doncs, per exemple, si l'usuari és un metge podrà inserir dades a l'historial, consultar el llistat dels pacients que té adjudicats i accedir a l'expedient mèdic dels pacients. En canvi, si es tracta d'un pacient que s'identifica enfront el sistema, aquest només ha de poder consultar les dades mèdiques incloses en el seu propi expedient mèdic.

A la base de dades cal tenir una taula amb la clau primària *Id_usuari* que contingui les dades del pacient o del metge i el seu certificat. En el cas que ens representa, tindrem una taula metge i una pacient amb les dades personals i mèdiques, i ambdues estaran relacionades amb una taula anomenada autenticació, la qual contindrà el DNI i el certificat corresponents al metge o pacient en qüestió. En endavant, el terme *Id_usuari* fa referència a aquest identificador per Document Nacional d'Identitat.

Els historials, corresponents a les dades de les visites efectuades per un metge sobre un determinat pacient, seran identificats de manera que es guarda una relació amb el pacient del qual es fa referència, acreditant que pertany a un únic usuari determinat.

Quan un metge o un pacient facin una consulta, cal que indiquin el seu *Id_usuari*, de manera que el gestor pugui trobar el seu certificat a la base de dades. És important recordar que el certificat s'utilitzarà per fer l'autenticació i també per saber a quin col·lectiu pertany. El camp Organizational Unit Name ens indicarà si l'usuari és un metge o un pacient, i el camp dnQualifier contindrà l'*Id_usuari* (DNI).

4.3.2. Autenticació

Descripció

Els usuaris del sistema, metges i pacients, s'autenticaran una única vegada al principi de cada connexió. Es pot dir així, que s'estableix una mena de sessió en la qual l'usuari podrà dur a terme un conjunt d'accions, havent-se autenticat enfront el sistema un únic cop. D'aquesta manera, es mantindrà un estat de la connexió amb l'objectiu que el gestor sigui coneixedor, en tot moment, de quin usuari correspon cada connexió. El manteniment de la connexió permetrà agilitzar i alleugerar les càrregues i connexions efectuades sobre el servidor de l'aplicació, convertint-se en un mètode eficient donat que només cal efectuar una única autenticació cada vegada que es vulgui iniciar una sessió. El gestor s'encarregarà de verificar que qualsevol usuari que vulgui accedir a l'aplicació passi o hagi passat per aquest procés d'autenticació.

El procés d'autenticació

Es parteix de la base de que cada usuari U s'identifica amb *Id_usuariU* i disposa d'una parella de claus (PU, SU) amb el corresponent certificat CertU. Aquest protocol, pot ser utilitzat per un metge o per un pacient.

En primer lloc, l'usuari genera un nombre aleatori Ni i, juntament amb el seu identificador d'usuari (Id_usuariU), confeccionarà un missatge. Aquest missatge, serà xifrat amb la clau pública del gestor PG i l'usuari li farà arribar a través de l'execució del servei.

Una vegada el gestor rep el missatge de l'usuari, aquest, el desxifrarà amb la seva clau privada SG, obtenint l'identificador d'usuari Id_usuariU i el número aleatori Ni. Seguidament, el gestor generarà un nou nombre aleatori, anomenat NG, i emmagatzemarà a la base de dades com a paràmetres temporals una entrada amb els valors Id_usuariU, el nombre Ni enviat per l'usuari, i el seu nou valor aleatori generat NG. A continuació, el gestor recuperarà de la base de dades el certificat que fa referència a l'usuari Id_usuariU, i, juntament amb els dos components aleatoris obtinguts (Ni i NG), el gestor elaborarà un missatge el qual xifrarà amb la clau pública PU de l'usuari en qüestió.

Aquest missatge de resposta del gestor, és enviat a l'usuari. En primera instància, l'usuari intentarà desxifrar el missatge amb la seva clau privada SU, obtenint els valors aleatoris corresponents (Ni' i NG). En cas de coincidir el valor Ni prèviament enviat per l'usuari, amb el valor Ni' enviat en la resposta del gestor, es podrà dir que l'usuari pot confiar amb el gestor donat que ha estat verificat com a entitat fiable, i l'usuariU, queda, doncs, parcialment autenticat. En cas contrari, si el Ni' enviat pel gestor no coincideixi amb l'enviat per l'usuari inicialment, es produirà un error en la validació de l'autenticació.

En el proper missatge, l'usuari enviarà el número NG' juntament amb les dades necessàries per a poder dur a terme l'opció del servei que vulgui realitzar (consulta d'historial, llistat de pacients o inserció de visita a l'historial). Abans de que es dugui a terme l'acció, el gestor verificarà que el nombre NG que va guardar a la base de dades coincideix amb el valor NG' que envia l'usuari U per tal d'aconseguir l'autenticació íntegra.

Resum del protocol

Protocol 1

1. PU realitza les operacions següents:
    a. Obtenir un valor de forma aleatòria: Ni
    b. Xifrar Ni i Id_usuariU amb la clau pública PG de G, PG[Ni, Id_usuariU]
    c. Enviar PG[Ni, Id_usuariU] a G
2. G realitza les operacions següents:
    a. Desxifrar PG[Ni, Id_usuariU] amb SG, i obtenir Ni i Id_usuariU
    b. Obtenir el certificat de U amb Id_usuariU. A partir del certificat s'obtindrà PU
    c. Obtenir un valor de forma aleatòria, NG
    d. Guardar a la BD els valors Ni i NG associats a Id_usuariU
    e. Xifrar Ni i NG amb la clau pública PU de U: PU[Ni, NG]
    f. Enviar PU[Ni, NG] a U

3. U realitza les operacions següents:

    a. Desxifrar PU[Ni, NG] amb clau privada SU i obtenir NG i Ni
    b. Si Ni = Ni'
        i. Autenticació parcial. L'usuari enviarà al gestor en la propera acció el missatge corresponent juntament amb el número aleatori NG desxifrat.
    c. Sinó es produeix un error en l'autenticació

4.3.3. Consulta d'un historial

Descripció

El protocol de consulta d'un historial pot ser utilitzat per un pacient o un metge. El protocol, requereix que l'usuari (metge o pacient) hagi realitzat prèviament el procés d'autenticació de forma correcta. Mitjançant un identificador d'usuari Id_usuari, el metge o pacient, indicarà quin és l'usuari sobre el qual es vol consultar les dades contingudes referents a l'expedient mèdic. Pel cas dels pacients, tan sols se'ls hi permet consultar el seu propi expedient clínic. En canvi, els metges podran consultar qualsevol expedient d'usuari, a excepció de les dades de visites que s'estimen com a privades, que només seran visibles per al metge, si el pacient és un usuari assignat a aquest.

El procés de consulta d'un historial

Per a poder executar aquest protocol, l'usuari ha d'haver dut a terme prèviament i de manera correcta el procés d'autenticació.

Primer, l'usuari crearà un missatge amb el valor aleatori del gestor NG (obtingut en el moment de l'autenticació), l'Id_usuariU i l'Id_usuari del qual vol consultar l'expedient mèdic. Un cop creat el missatge, es xifrarà amb la clau pública del gestor PG i s'enviarà per a què el rebi el gestor.

El gestor, en el moment de rebre la sol·licitud xifrada, la desxifra amb la seva clau privada SG i obté NG' enviat per l'usuari, l'Id_usuariU i l'Id_usuari del qual es vol recopilar la informació. A continuació el gestor recuperarà de la base de dades el valor NG que es va guardar a l'inici de la sessió amb l'usuari amb Id_usuariU, i comprovarà que els valors aleatoris NG i NG' coincideixen entre ells. En cas afirmatiu, el gestor procedeix amb l'execució del protocol.
A partir d'aquest moment, el gestor comprovarà el perfil de l'usuari amb Id_usuariU i es poden preveure tres possibles situacions:

- Si l'usuari Id_usuariU es tracta d'un metge, caldrà després verificar si Id_usuari es tracta d'un pacient assignat al metge Id_usuariU. Si és pacient del metge Id_usuariU, es recopilarà tota la informació del pacient Id_usuari, tant si les dades de l'historial de visites són de caràcter privat o públic.

- Si l'usuari Id_usuariU es tracta d'un metge però el pacient Id_usuari no és un pacient assignat del metge Id_usuariU, tan sols es recolliran aquelles dades considerades públiques de les visites de l'historial del pacient.

- Si l'usuari Id_usuariU es tracta d'un pacient, l'Id_usuari correspondrà amb Id_usuariU (es tracta del propi pacient) donat que els pacients tan sols poden consultar els seu propi historial mèdic.

Seguidament, el gestor s'encarregarà de recopilar tota aquella informació referent a l'historial de l'usuari amb Id_usuari demanat tenint en compte la situació donada, i confeccionarà el missatge de resposta per a l'usuari amb Id_usuariU.

Per a cada entrada de les visites a la base de dades associades a al pacient amb Id_usuari, el gestor verificarà la signatura corresponent SG[M] amb la seva clau asimètrica SG abans d'insertar-la al missatge de resposta. Tanmateix, el gestor verificarà el darrer número de sèrie a partir de la signatura SG[X] corresponent. Abans de crear el missatge de resposta, el gestor també verificarà la integritat de les dades personals del pacient amb Id_usuari del qual es vol consultar l'historial mèdic. Un cop acabat el missatge de resposta, aquest primer de tot és signat amb la clau asimètrica del gestor SG, i es xifra el missatge i la signatura del missatge M i S[M] amb la clau pública PU de l'usuari amb Id_usuariU, i se li retorna PU[M, S[M]].

Una vegada l'Id_usuariU rep la resposta del gestor, aquest desxifra el missatge amb la seva clau privada SU i obté el missatge M i la seva signatura SG[M]. Immediatament, l'usuari verifica la signatura per a validar la integritat del missatge rebut per part del gestor. A continuació, es passarà a verificar per a cada entrada de l'historial de visites, la signatura del gestor, la signatura del metge corresponent i, per últim, es verificarà la seqüència d'aquestes. Si la verificació és satisfactòria, l'Id_usuariU obtindrà les dades referents a l'expedient del pacient demanat de l'Id_usuari. Tanmateix, per a cada entrada a l'historial de visites, l'usuari comprovarà la signatura del gestor, la signatura del metge i la seqüència de visites corresponent. D'aquesta manera, s'assegurarà la integritat de cada visita i l'autoria de la persona que la va realitzar.

Finalment, una vegada acabat el procés de verificació de les signatures, es podrà estar seguir de la informació obtinguda i es procedirà a la seva presentació.

Resum

Protocol 2

1. U realitza les operacions següents:

    a. Xifrar N_G, Id_usuariU, Id_usuari amb la clau pública P_G de G, P_G [N_G, Id_usuariU, Id_usuari]. El missatge de consulta indicarà que es vol consultar l'historial de l'usuari identificat amb l'Id_usuari
    b. Enviar P_G[N_G, Id_usuariU, Id_usuari] a G

2. G realitza les operacions següents:

    a. Desxifrar P_G[N_G, Id_usuariU, Id_usuari] amb la clau privada S_G i obtenir N_G', Id_usuariU, Id_usuari
    b. Recuperar N_G de la BD. En el pas 2 de l'autenticació, 1 N_G i Ni han estat guardats a la BD
    c. Si NG' = NG fer:
        i. Si (Id_usuariU = Id_usuari) o (Id_usuariU és metge) fer:
            1. Executar el Procedure 1 amb Id_usuari i PU, i obtenir PU[H]
            2. Enviar PU[H] a U

ii. Sinó retornar error
d. Sinó retornar error

3. U realitza les operacions següents:

    a. Executar el Procedure 2 amb PU[H], i obtenir H
    b. Mostrar H

Procedure 1 (Id_usuari, PU)

El gestor G utilitza el Procedure 1 per trobar l'historial que se li ha demanat i desxifrar-lo amb la clau de l'usuari que es vol consultar.

1. Buscar l'historial H corresponent a Id_usuari
2. Si (Id_usuariU és metge però Id_usuari no és pacient assignat), fer:
    a. Desxifrar la part de H que està xifrada utilitzant la clau privada SG de G
    b. Obtenir H amb les visites que no són privades
3. Sinó, fer:
    a. Desxifrar la part de H que està xifrada utilitzant la clau privada SG de G
    b. Obtenir H
4. Comprovar que la signatura SG[X] coincideix amb el darrer número de l'última visita
5. Xifrar H amb la clau pública PU, PU[H]
6. Retornar PU[H]

Procedure 2 (PU[H])

Un usuari utilitza el Procedure 2 per tal de desxifrar un historial enviat pel gestor G i verificar que l'historial és correcte.

1. Desxifrar PU[H] amb clau privada SU de U: SU[PU[H]]
2. Per a cada entrada de l'historial H que està signada, fer:
    a. Verificar la signatura digital de M
    b. Verificar la signatura digital de G
    c. Verificar la seqüència (de X i T)
3. Retornar H

4.3.4. Consulta dels pacients assignats a un metge

Descripció

Una operació típica d'un metge és buscar l'historial d'un dels seus pacients. Amb el protocol 3, un metge pot obtenir el llistat dels seus pacients. A la descripció només s'envien els identificadors d'usuaris. Aquesta és la informació, els identificadors, és mínima per recuperar un historial mèdic d'un pacient determinat per l'identificador Id_usuari.

El protocol per a la consulta dels pacients assignats d'un metge, només podrà ésser executat per un usuari que segueixi un perfil de metge. El protocol, requereix que l'usuari (metge) hagi realitzat prèviament el procés d'autenticació de forma correcta.

El procés de consulta dels pacients assignats a un metge

Per a poder executar aquest protocol, l'usuari ha d'haver dut a terme prèviament i de manera correcta el protocol d'autenticació.

Primer, l'usuari crearà un missatge amb el valor aleatori del gestor NG obtingut en el moment de l'autenticació i l'identificador de l'usuari Id_usuariU. Un cop creat el missatge, es xifrarà amb la clau pública del gestor PG i s'enviarà per a què el rebi el gestor.

El gestor, en el moment de rebre la sol·licitud xifrada, la desxifra amb la seva clau privada SG i obté NG' enviat per l'usuari i l'Id_usuariU. A continuació el gestor recuperarà de la base de dades el valor NG que es va guardar a l'inici de la sessió amb l'usuari amb Id_usuariU, i comprovarà que els valors aleatoris NG i NG' coincideixen entre ells. En cas afirmatiu, el gestor procedeix amb l'execució del protocol.

Seguidament, el gestor recuperarà de la base de dades els pacients assignats al metge que sol·licita el llistat i que és identificat per Id_usuariU. A continuació, un cop recuperada la llista dels pacients, el gestor la signarà amb la seva clau privada i confeccionarà un missatge M on inclourà el llistat dels pacients i la signatura d'aquest llistat. Una vegada s'ha elaborat el missatge, el gestor obtindrà la clau pública del metge PU per a xifrar el missatge M, obtenint PU[[pacient1, pacient2, ... , pacientN], SG[pacient1, pacient2, ..., pacientN]] i li retornarà a l'usuari metge.

L'usuari Id_usuariU rep el missatge PU[[pacient1, pacient2, ... , pacientN], SG[pacient1, pacient2, ..., pacientN]] xifrat pel gestor, i el desxifra amb la seva clau privada SU. Així, l'usuari, obté el llistat dels pacients que té assignats i la signatura d'aquest llistat. En aquest moment, l'usuari procedeix a comprovar la integritat i l'autoria del llistat mitjançant la verificació de la signatura del llistat amb la clau pública del gestor.

En cas de que la verificació del llistat sigui satisfactòria, es presentarà per a l'usuari el llistat dels pacients corresponents que té assignats.

Resum

Protocol 3

1. U realitza les operacions següents:
    a. Xifrar N_G i Id_usuariU amb la clau pública P_G de G, P_G [N_G, Id_usuariU]. El missatge de consulta indicarà que es vol consultar el llistat de pacients per a l'usuari identificat amb l'Id_usuariU
    b. Enviar P_G[N_G, Id_usuariU] a G
2. G realitza les operacions següents:
    a. Desxifrar PG[NG, Id_usuariU]] amb la clau privada SG, i obtenir NG' i llista_pacients
    b. Recuperar NG de la Base de Dades (en el pas 2 de l'autenticació, NG i Ni han estat guardats a la Base de dades)
    c. Si NG' = NG fer:
        i. Si Id_usuariU és metge, fer:
            1. Executar el Procedure 3 amb Id_usuariU i PU per obtenir: PU[{Id_usuari1, ... , Id_usuarin},SG[{Id_usuari1, ..., Id_usuarin}]]

2. Enviar a U PU[{Id_usuari1, ... , Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]]

3. U realitza les operacions següents:

    a. Executar el Procedure 4 amb PU[{Id_usuari1, ... , Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]]
    b. Mostrar {Id_usuari1, ... , Id_usuarin}

Procedure 3 (Id_usuari, PU)

Amb el Procedure 3 el gestor G obté el llistat dels pacients assignats al metge Id_usuari.

1. Cercar a la Base de Dades els pacients assignats al metge Id_usuari, obtenint {Id_usuari1, ..., Id_usuarin}
2. Signar {Id_usuari1, ..., Id_usuarin} amb la clau privada SG de G, SG[{Id_usuari1, ..., Id_usuarin}]
3. Xifrar {Id_usuari1, ..., Id_usuarin} i SG[{Id_usuari1, ..., Id_usuarin}] amb la clau pública de l'usuari de Id_usuari PU, obtenint: PU[{Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]]
4. Retornar PU[{Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]]

Procedure 4 (PU[{Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]])

El Metge utilitza el Procedure 4 per obtenir la llista dels seus pacients i verificar que ha estat generada pel Gestor G.

1. Desxifrar PU[{Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]] amb la clau privada SU de U, SU[PU[{Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]]] i obtenir: {Id_usuari1, ..., Id_usuarin}, SG[{Id_usuari1, ..., Id_usuarin}]
2. Verificar la signatura digital SG[{Id_usuari1, ..., Id_usuarin}] amb la clau pública PG de G
3. Si la verificació anterior és correcta, mostrar {Id_usuari1, ..., Id_usuarin}

4.3.5. Inserció de les dades a l'historial mèdic

Descripció

El protocol per a la per a la inserció de les dades referents a una nova visita, només podrà ésser executat per un usuari que segueixi un perfil de metge. El protocol, requereix que l'usuari (metge) hagi realitzat prèviament el procés d'autenticació de forma correcta.

Aquest protocol està pensat únicament per afegir una nova visita V a l'historial de visites per al pacient Id_usuari. El gestor un cop rep una visita V d'un pacient P verifica que ha estat signada pel metge M assignat al pacient. A continuació afegeix la visita a l'historial H xifrant la visita. Per garantir que l'ordre de visites no es modifica, s'afegeix a cada visita una marca temporal T i un número de sèrie X. Amb aquestes dades es pot saber l'instant de te la visita i l'ordre que han seguit. La visita V, el temps T i el número de sèrie són signats pel gestor G. Si un atacant elimina un registre al mig de l'historial, es detectarà perquè hi haurà un salt en un número a la sèrie de les visites. L'atacant no podrà refer la seqüència sense la clau privada del gestor G. Suposem que aquesta clau està ben protegirà (és la base per a garantir la seguretat del sistema). A continuació, el gestor G xifra les dades de la visita amb la seva clau pública i ho guarda a la BD. Si un atacant

accedeix a la BD no pot veure les dades confidencials referents a l'historial. Finalment, afegeix a la base de dades una signatura digital de quin és l'últim número de la sèrie X de l'historial H. Si un atacant elimina l'última visita es detectarà perquè hi haurà un salt entre l'última visita i el número de sèrie X signat.

El procés d'inserció d'una visita

Per a poder executar aquest protocol, l'usuari ha d'haver dut a terme prèviament i de manera correcta el protocol d'autenticació. Tanmateix, en aquest protocol es suposa que prèviament a la inserció de les dades, el metge M ha d'haver consultat l'historial del pacient P, i per tant coneix Id_usuari associada.

Primer, el metge M recopilarà el valor aleatori NG que havia estat enviat pel gestor en el moment de l'autenticació i les dades de la visita (que inclou l'Id_usuariU del metge i l'Id_usuari del pacient) a inserir per al pacient P. Quan es disposin de les dades que confeccionen la visita, aquesta serà signada amb la clau privada del metge M. Un cop es disposen de totes aquestes dades (NG, les dades que configuren la visita i la signatura per part del metge d'aquesta), es procedeix a crear el missatge que s'enviarà al gestor. Abans d'ésser enviat, aquest missatge es xifrarà amb la clau pública del gestor PG. Una vegada xifrat, s'enviarà per a què el rebi el gestor.

El gestor, en el moment de rebre el missatge xifrat, el desxifra amb la seva clau privada SG i obté, en primera instància, el valor aleatori NG' enviat per l'usuari i l'Id_usuariU. A continuació el gestor recuperarà de la base de dades el valor NG que es va guardar a l'inici de la sessió per l'usuari amb Id_usuariU, i comprovarà que els valors aleatoris NG i NG' coincideixen entre ells. En cas afirmatiu, el gestor procedeix amb l'execució del protocol.

Seguidament, el gestor obté la visita i la signatura d'aquesta efectuada per part del metge. De la visita, el gestor obté l'Id_usuariU metge i Id_usuari referent al pacient. Primer comprova que l'Id_usuariU és metge i, després, verifica que Id_usuari és pacient del metge amb Id_usuariU. Si tot és correcte, a continuació, verifica la signatura de la visita corresponent del metge amb la clau pública del certificat del metge que es recuperarà de la base de dades. Una vegada s'ha verificat la signatura, el gestor obté l'instant de temps actual T i recupera el número de sèrie X de l'última visita de l'historial H, aquest darrer paràmetre es troba emmagatzemat a la base de dades. En aquest moment, el gestor incrementa en una unitat el número de sèrie X. A continuació, el gestor signa la visita V, SM[V], T + Id_usuari i X actual + Id_usuari amb la clau privada SG del gestor. Finalment, el gestor xifra V i SM[V] amb la clau pública SG i insereix a la taula visita corresponent a la base de dades els següents valors: PG[V,SM[V]], signatura SG[V,SM[V]], X, signatura SG[X + Id_usuari], T, signatura SG[T + Id_usuari]. Addicionalment, el gestor insereix la signatura SG[X + Id_usuari] a la BD, per tal de poder disposar de la signatura del número de sèrie actual per a comprovar i validar, en el moment que sigui necessari, quin és el darrer número de sèrie.

Resum

Protocol 4

1. M realitza les operacions següents

    a. Obtenir les dades de la visita V. La visita inclourà Id_usuariM i Id_usuariP com a identificadors respectius per al metge i pacient sobre els quals es produeix la visita en qüestió
    b. Signar V amb la clau privada SM de M, SM[V]

c. Xifrar NG, V i SM[V] amb la clau pública PG de G, PG[NG, V, SM[V]], indicaant que es vol afegir V a l'historial del pacient P (l'Id_usuariP es troba dins de les dades de la visita V)
d. Enviar PG[NG, V, SM[V]] a G

2. G realitza les operacions següents:
    a. Desxifrar PG[NG, V, SM[V]] amb la clau pirvada SG i obtenir NG', Inserir_visita, V i SM[V].
    b. Recuperar NG de la BD. En el pas 2 del procés d'autenticació, NG i Ni han estat guardats a la BD.
    c. Si NG'=NG fer:
        i. Obtenir Id_usuariP a partir de V
        ii. Obtenir Id_usuariM a partir de V
        iii. Verificar que Id_usuariM és metge
        iv. Verificar que Id_usuariP és un pacient assignat a Id_usuariM
        v. Si les verificacions anteriors són correctes, fer:
            1. Verificar la signatura digital SM[V] amb la clau pública PM
            2. Obtenir l'instant de temps actual T
            3. Obtenir el número de sèrie X de l'última visita de l'historial H del pacient Id_usuariP
            4. Incrementar en una unitat X, X+1
            5. Signar V, SM[V], (T i Id_usuariP), (X+1 i Id_usuariP), amb clau privada SG de G: SG[V, SM[V]], SG[T i Id_usuariP], SG[X+1 i Id_usuariP]
            6. Xifrar V i SM[V] amb la clau pública SG de G, PG[V,SM[V]]
            7. Guardar a la BD PG[V, SM[V]], T, X+1, SG[V, SM[V]], SG[T, Id_usuariP] i SG[X+1, Id_usuariP]
        vi. Sinó retornar error
    d. Sinó retornar error

4.3.6. Finalitzar sessió

Descripció

Els usuaris del sistema, metges i pacients, s'autentiquen enfront al sistema i mantenen una sessió d'usuari. Així, l'autenticació de l'usuari es realitza només una vegada al principi de la sessió i, per tant, es fa necessari finalitzar la sessió, és a dir, esborrar de la base de dades aquelles dades que permeten al Gestor associar l'usuari que feia una petició.

Per a poder executar aquest protocol, doncs, l'usuari (metge o pacient) ha d'haver dut a terme prèviament i de manera correcta el protocol d'autenticació.

El procés de finalitzar sessió

Aquest protocol, pot ser utilitzat per un metge o per un pacient. Inicialment, l'usuariU identificat per Id_usuari, manté en memòria el valor aleatori NG enviat pel Gestor quan es va autenticar en el sistema. L'usuari, recupera el valor NG i junt amb el seu identificador, els xifra amb la clau pública del Gestor obtenint PG[NG, Id_usuari]. A continuació, envia el missatge al Gestor.

Un cop el Gestor rep la sol·licitud per finalitzar la sessió, aquest desxifra el missatge PG[NG, Id_usuari] amb la clau privada SG i obté NG' i Id_usuari. A continuació procedeix a l'eliminació de les dades a la BD que identifiquen a l'usuari amb la seva sessió corresponent. Si l'esborrat és

correcte, la sessió finalitza correctament. En cas contrari, si el Gestor no troba els paràmetres a la BD, es produeix un error.

Resum del protocol

Protocol 1

1. PU realitza les operacions següents:

    a. Obtenir el valor aleatori enviat inicialment pel gestor: NG
    b. Xifrar NG i Id_usuari amb la clau pública PG de G, PG[NG, Id_usuari]
    c. Enviar PG[NG, Id_usuari] a G

2. G realitza les operacions següents:

    d. Desxifrar PG[NG, Id_usuari] amb SG, i obtenir NG' i Id_usuari
    e. El gestor procedeix a l'eliminació de les dades de sessió de la BD a partir de l'identificador de l'usuari U i del nombre aleatori NG' guardats a la BD en el moment de l'autenticació.
    f. Si NG = NG' i NG' pertany a Id_usuari a la BD
        i. Eliminació de la sessió i dels paràmetres Ni, NG i Id_usuari
        ii. Sessió finalitzada per a l'usuari amb Id_usuari
    g. Sinó retornar error

5. Arquitectura del sistema

5.1. Introducció

Abans d'implementar, s'ha dut a terme un estudi dels requeriments en el capítol 2 i en el qual es definien i es detallaven les necessitats sistema envers als actors que interaccionen, les accions i els serveis que ha de proporcionar als usuaris, la gestió i configuració de la informació que és tractada, i, els requeriments de seguretat que s'han d'acomplir.

Després, s'han confeccionat els esquemes i protocols criptogràfics per cada acció o servei identificat, els quals han de suposar la base per a conduir la implementació de l'aplicació del propi sistema.

Un cop arribat a aquest punt, es procedeix a definir l'arquitectura i l'estructura que forma l'entorn en la qual el sistema s'ha d'establir i que esdevindrà una guia per a poder concebre de manera adequada la implementació i desenvolupament de les parts corresponents.

Tanmateix, es detallen i documenten aquells aspectes i aquelles decisions preses que s'han tingut en compte i que són necessaris per a poder dur a terme la implementació del sistema.

5.2. Disseny de l'aplicació

El sistema es concep dins d'un model d'arquitectura lògica que segueix un disseny de programació estructurat en tres parts. Així, es pot dir que aquesta divisió de la implementació correspondrà a la formació d'una part referent a la presentació, una altra referent al cos o domini i, una darrera part, encarregada a gestionar i emmagatzemar les dades.

- Presentació: correspon a les classes que confeccionarien la interfície gràfica d'usuari, la funció de la qual destacaria per rebre i atendre els esdeveniments emesos en forma de

peticions per part de l'usuari. Aquests esdeveniments són transformats en forma de crides i execucions de les accions i mètodes, comunicant el resultats obtinguts a l'usuari.

- Domini o implementació: s'encarrega de rebre els esdeveniments originaris de la part corresponent a la interfície gràfica. Determinarà la validesa dels esdeveniments i efectuarà les accions demanades, realitzant les operacions necessàries

- Gestió de les dades: s'encarrega de donar servei a les peticions que rep i de la gestió de les dades (lectura, inserció, actualització i esborrat).

La part corresponent a la presentació o interfície gràfica d'usuari es podrà veure al capítol 9 i, la gestió de les dades, es defineix al capítol 8. La part corresponent a la implementació es comença a detallar a partir d'aquest mateix capítol.

5.3. Aplicatius

Tot seguit, a continuació, es defineixen les funcions dels aplicatius considerats per a la seva implementació.

Aplicatiu pacient

L'aplicatiu pacient és el programari que utilitzarà un pacient per accedir de forma segura al sistema i on s'ha de poder dur a terme un conjunt d'accions relacionades. Les funcionalitats que l'aplicatiu pacient ha d'atendre, són les següents:

- Autenticar al pacient enfront al sistema
- Realitzar una consulta del seu expedient mèdic
- Abandonar de forma segura el sistema

Aplicatiu Metge

L'aplicatiu metge és el programari que utilitzarà un metges per accedir de forma segura al sistema i on s'ha de poder dur a terme un conjunt d'accions relacionades. Les funcionalitats que l'aplicatiu metge ha de garantir, són les següents:

- Autenticar al metge enfront al sistema
- Llistar els pacients assignats
- Realitzar una consulta de l'historial d'un dels pacients
- Inserir les dades d'una nova visita a l'historial d'un dels pacients assignats del metge
- Abandonar de forma segura el sistema

5.4. Implementació

A la figura 5-1 es pot veure el disseny UML corresponent al diagrama de classes inicial del sistema. Cal destacar tres classes principals: Metge, Pacient i Gestor.

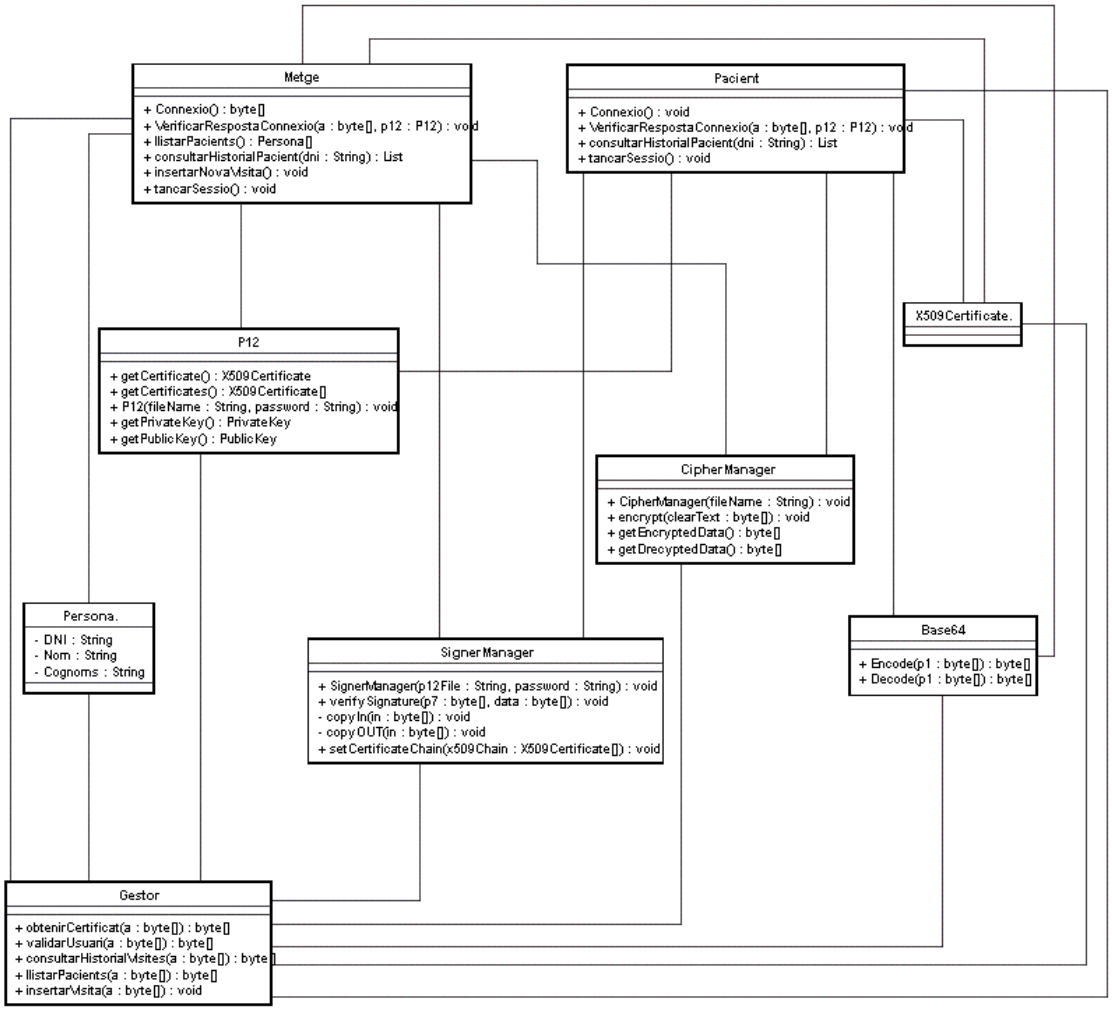

Figura 5-1. Diagrama de classes

Seguidament, es fa un petit incís sobre cadascuna de les classes que apareixen en el diagrama *UML*:

- Gestor: és la classe representant del servidor i implementa el conjunt de mètodes necessaris per garantir el compliment de les següents funcionalitats donant, alhora, una resposta segura:
    - Autenticar els usuaris
    - Consulta d'historial de pacients
    - Consulta de llistat de pacients
    - Inserció de visites
    - Finalització de sessions d'usuari

- Metge: és la classe encarregada de dur a terme totes les peticions als serveis corresponents dels metges.

- Pacient: és la classe encarregada de dur a terme totes les peticions als serveix corresponents dels pacients.

- SignerManager: classe encarregada del tractament de signatures. Permet realitzar signatures digitals i obtenir la verificació d'aquestes.

- CipherManager: classe encarregada del tractament de les operacions de xifratge i desxifratge de dades.

- P12: classe encarregada de l'accés a arxius *PKCS#12* i oferir mètodes per accedir i manipular aquests magatzems de dades, els quals contenen el parell de claus i el certificat corresponent.

- Base64: classe que permet codificar i descodificar utilitzant un sistema de numeració posicional que usa 64 com a base.

- X509Certificate: és una classe auxiliar per a poder treballar amb els certificats digitals dels usuaris.

- Persona: classe auxiliar per a poder estructurar certa informació relacionada amb mètodes com el llistat de pacients.

Posteriorment a la implementació del diagrama de classes de la Figura 5-1, es considera la possibilitat d'afegir una classe que la qual contingui certs mètodes comuns per a facilitar operacions com, per exemple, l'obtenció d'un número aleatori binari, el desxifratge i xifratge d'informació utilitzant CipherManager, la verificació de la signatura i el procés de signatura amb SignerManager, descodificació en format base 64, etc.

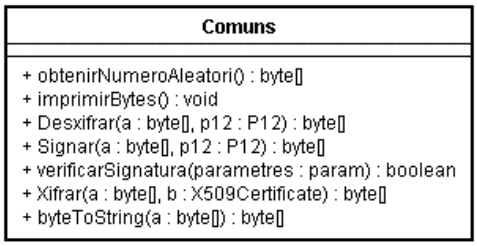

Figura 5-2. Classe Comuns

En els propers capítols es podrà veure com el diagrama de classes inicial es va adaptant d'en mica en mica amb noves classes i implementacions, a mida que es van incorporant noves característiques funcionals al sistema.

En primer lloc, s'analitzarà el desenvolupament per a estructurar els missatges amb els que es comuniquen els usuaris i el sistema, seguint el format proporcionat per *XML*.

També, tal i com es va proposar en els objectius del projecte, s'incorporaran noves implementacions per a poder utilitzar la tecnologia *Remote Method Invocation* de *Java* per a la realització de la comunicació entre les aplicacions.

D'altra banda, s'utilitzarà una base de dades per a permetre emmagatzemar les dades i continguts relacionats amb els pacients, metges i els historials mèdics. D'aquesta manera, s'inclouran noves adaptacions al sistema.

Finalment, es detallaran les necessitats per a la implementació de la part corresponent a la interfície gràfica d'usuaris.

6. XML

6.1. Introducció

L'XML, corresponent a les sigles en anglès a *Extensible Markup Language* (llenguatge de marques extensible), és un metallenguatge extensible d'etiquetes desenvolupant pel World Wide Web Consortium (W3C).

XML és un llenguatge que permet jerarquitzar i estructurar la informació i descriure els continguts dins d'un propi document, així com, també, la reutilització de les parts d'aquest. XML no ha nascut només per a la seva aplicació a Internet, sinó que es proposa com a un estàndard per a l'intercanvi d'informació estructurada entre diferents plataformes. La informació estructurada presenta diferents continguts (text, imatges, àudio, etc) i diferents formes en la que la seva aplicació pot ser de gran utilitat: es pot utilitzar per a base de dades, editors de text, fulls de càlcul i per a moltes altres aplicacions diverses.

XML és una tecnologia relativament senzilla que té al seu voltant altres que la complementen i la fan notablement més extensa, a més de proporcionar-li unes possibilitats molt més grans. Inicialment va sorgir com un llenguatge de marques per a substituir a l'HTML. Ambdós llenguatges provenen del SGML, un llenguatge de marques estàndard per a la descripció formal i de contingut de documents. Es tracta d'una simplificació i adaptació de l'experimentat SGML, i permet definir la gramàtica de llenguatges específics. No es tracta d'un llenguatge en particular, sinó una manera de definir llenguatges per a diferents necessitats.

Fou concebut des del camp empresarial, donat que HTML era un llenguatge poc potent per a suportar de manera eficaç i de forma massiva la producció de negocis virtuals. En l'actualitat, l'XML té un paper molt important, ja que permet la comptabilitat entre una gran varietat de sistemes, permetent de compartir informació d'una manera segura, fiable i fàcil.

Estructura d'un document XML

Els documents XML es basen en documents de text pla en els que s'utilitzen etiquetes per delimitar els elements continguts en un document XML. XML defineix aquestes etiquetes en funció del tipus de dades que s'està escrivint i no de l'aparença final que tindran, com ho fan els documents HTML. Tanmateix, donada l'extensibilitat del llenguatge, permet definir noves etiquetes i ampliar les existents.

Una etiqueta correspon a una marca feta en el document XML, que senyala una porció d'aquest com un element, formant un contingut d'informació amb un sentit clar i definit.

Anomenem documents XML ben formats (*well-formed*) a aquells que compleixen amb totes les definicions bàsiques de format i, per tant, poden ésser analitzats correctament per qualsevol analitzador sintàctic que compleixi amb la norma.

Els documents han de seguir una estructura estrictament jeràrquica pel què respecta a les etiquetes que delimiten els seus elements. Una etiqueta ha d'estar correctament inclosa dins d'una altra. Els elements amb contingut han d'estar correctament tancats.

Els documents XML només permeten un element arrel del què la resta en formin part, és a dir, només poden tenir un element inicial.

Les construccions tals com etiquetes o referències d'entitat s'anomenen marques, i formen part del document que el processador XML espera entendre. La resta del document entre marques són les dades comprensibles per a les persones.

Avantatges del XML

De les seves propietats i característiques del llenguatge, es poden percebre ràpidament els avantatges en l'ús d'aquesta tecnologia:

- Està basat en text.
- Suporta *Unicode*, permetent així la comunicació amb pràcticament tots els idiomes escrits.
- Pot representar les estructures de dades del món dels ordinadors: registres, llistes i arbres.
- L'estricta sintaxi, i els requeriments d'anàlisi sintàctica d'aquesta tecnologia, fan que els algorismes d'anàlisi hagin de ser extremadament simples, eficients i coherents.
- És extensible, el què permet que un cop creat el llenguatge, sigui possible expandir-lo gràcies a la creació de noves etiquetes.
- Ajuda a mantenir la compatibilitat entre versions més antigues i de més recents d'un determinat document.
- XML és adequat per a ésser utilitzat coma format per a l'emmagatzematge i processat de documents, tant *online* com *offline*.
- Està basat en els estàndards internacionals.
- Permet la validació emprant llenguatge esquemàtics.
- L'estructura jeràrquica és convenient per a la majoria de tipus de documents.
- Els arxius són creats com a text pla, cosa que els fa menys restrictius que d'altres propietaris.
- Un fragment de l'element d'un document ben format de XML és també un document ben format XML.

Apunt

Una restricció del format en documents XML és que ha d'estar en format text. Si les aplicacions han d'intercanviar-se algun missatge que hagi de contenir dades binàries, com per exemple és el cas d'una signatura digital o un missatge xifrat, s'hauran de representar com si fossin text. Per a aquest projecte, aquest fet ha fet necessari codificar aquest tipus de dades en format base 64.

D'altra banda cal destacar que, en la creació dels missatges XML de l'aplicació, s'ha utilitzat una codificació de caràcters corresponent a UTF-8. És per aquest motiu que es recomana no afegir missatges amb accents o amb caràcters que no suportin aquesta codificació. Malgrat aquest petit detall, s'ha vist que és possible canviar la configuració de la codificació de caràcters tal i com s'ha esmentat anteriorment.

6.2. Documents XML utilitzats

En el projecte s'ha fet ús de la tecnologia XML per a expressar els continguts d'informació que es produeixen en les transferències de les comunicacions remotes que s'estableixen durant l'execució dels protocols criptogràfics.

Aquests documents XML, corresponen tant a les peticions realitzades pels propis usuaris en les quals es confeccionen sol·licituds per a executar una determinada acció o servei determinat, com per les respostes generades del gestor en base a les peticions d'informació rebudes i a les quals cal donar una resposta.

Així, cada cop que es produeix un intercanvi d'informació entre el client i el gestor, només s'enviarà el document XML tenint en compte la fase del protocol en la que s'estigui duent a terme la comunicació. Per a facilitar la gestió sobre la confecció, manipulació i accés dels missatges que s'intercanvien entre les diferents parts, s'ha creat una classe anomenada XML (veure secció 5.3.) la qual ofereix els mecanismes necessaris per generar i llegir la informació i les dades requerides en cada possible situació.

Addicionalment, cal comentar que per a fer possible la implementació dels mètodes d'accés i generació dels documents *XML*, s'ha fet ús de la llibreria en Java anomenada JDOM (*Java Document Object Model*). Aquesta és una API pública pensada específicament per al processament de documents *XML* en Java, permetent el reconeixement sintàctic (*parser*) per a la creació, manipulació i serialització de documents *XML*.

Seguidament, es detallarà una relació d'exemple dels principals tipus de documents *XML* emprats per a l'intercanvi de dades entre els usuaris i gestor durant l'execució dels protocols que configuren el sistema.

6.2.1 Document XML de referència

Tots els missatges confeccionats en format *XML* per a les diferents accions que es poden dur a terme, presenten una estructura semblant a la que segueix. Es pot dir que, aquest, és el document bàsic.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
  <Missatge>
  </Missatge>
  <MissatgeSignat>
  </MissatgeSignat>
</Document>
```

Figura 6-1. Document XML de referència

6.2.2 Document XML de petició d'autenticació

El document *XML* referent a la petició d'autenticació inclou el número aleatori Ni que genera l'usuari i el seu DNI el qual l'identifica. El número aleatori generat, tal i com s'ha comentat anteriorment i com també succeeix en totes les marques que contenen dades referents a signatures i de xifratge, ha de ser transformat de byte a base 64 per a poder ésser inserit com a text i passar a formar part del document.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
  <Missatge>
      <NumeroAleatoriUsuari>
            gl7B3W92CUg5ZpEdzGas91zVXen4mw1aOQqB7xqSCU0=
      </NumeroAleatoriUsuari>
      <UsuariDNI>33333333-A</UsuariDNI>
  </Missatge>
</Document>
```

Figura 6-2. Document XML de petició d'autenticació

6.2.3 Document XML de resposta d'autenticació

En el missatge en format *XML* que emet el gestor com a resposta de la petició d'autenticació de l'usuari, disposarà del número aleatori generat per l'usuari Ni i un número aleatori que genera el gestor NG.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
  <Missatge>
      <NumeroAleatoriUsuari>
            /acG///ctAJLoU/m2YbD1mFmP6pbB1FOH5RgWunOXqw=
      </NumeroAleatoriUsuari>
      <NumeroAleatoriGestor>
            40Ptao1hxxu/qetD7DULqF3877jYtZ8EtdR2gHU3Q/s=
      </NumeroAleatoriGestor>
  </Missatge>
</Document>
```

Figura 6-3. Document XML de resposta d'autenticació

6.2.4 Document XML de petició de consulta d'historial

En el moment que un usuari sol·licita la consulta d'un historial mèdic, es genera un document *XML* composat pel número aleatori del gestor NG, el DNI del pacient sobre el qual es vol realitzar la consulta, i, finalment, el DNI que emet la petició.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
  <Missatge>
      <NumeroAleatoriGestor>
            S/vuxkLHlsiWrE8TMW68C/O5mpjGaMvWxeyUAKABfxE=
      </NumeroAleatoriGestor>
      <PacientDNI>66666666-A</PacientDNI>
      <UsuariDNI>33333333-A</UsuariDNI>
  </Missatge>
</Document>
```

Figura 6-4. Document XML de petició de consulta d'historial

6.2.5 Document XML de resposta de consulta d'historial

El document *XML* que forma la resposta del gestor i en el qual es pot veure un recull de les dades que conformen l'historial del pacient, conté un conjunt nombrós de marques amb la seva informació respectiva.

En primer lloc, es pot observar un conjunt de dades que fan referència a les dades personals de l'usuari. Tanmateix, s'inclou un grup del que podríem anomenar dades mèdiques bàsiques que contenen informació respecte al pes, alçada i grup sanguini del pacient. D'altra banda, es pot veure el recull de visites del pacient, estructurades de tal manera que cadascuna conté el seu número de sèrie i la signatura d'aquest, la marca temporal i la seva signatura, el conjunt de dades pròpies de

la visita, la signatura del metge i la signatura del gestor. Per últim, es pot observar que existeix una signatura del gestor que fa referència a tot el missatge per a garantir la seva integritat.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
   <Missatge>
      <DadesPersonals>
            <DNI>66666666-A</DNI>
            <Nom>Laura</Nom>
            <Cognoms>Girona</Cognoms>
            <Adreca>c/Saragossa num. 267, 2n 4a</Adreca>
            <Poblacio>Barcelona</Poblacio>
            <Telefon>934569898</Telefon>
            <DataNaixement>06/09/1974</DataNaixement>
      </DadesPersonals>
      <DadesMediques>
            <Pes>66kg</Pes>
            <Alcada>171cm</Alcada>
            <GrupSanguini>AB-</GrupSanguini>
      </DadesMediques>
      <Visites>
            <MissatgeVisita>
                  <numSerie>1</numSerie>
                  <SignaturaX> ...</SignaturaX>
                  <Data>03/01/2008 18:06</Data>
                  <SignaturaT> ...</SignaturaT>
                        <Visita>
                              <Metge>33333333-A</Metge>
                              <Pacient>66666666-A</Pacient>
                              <Diagnostic>Grip</Diagnostic>
                              <Tractament>
                                    Dafalgan 1 gram.- 3 cops al dia
                              (durant una set.)
                              </Tractament>
                              <Evolucio>-</Evolucio>
                              <Privat>0</Privat>
                        </Visita>
                  <SignaturaMetge>...</SignaturaMetge>
                  <SignaturaGestor>...</SignaturaGestor>
            </MissatgeVisita>
            <MissatgeVisita>
            ...
            </MissatgeVisita>
      </Visites>
  </Missatge>
  <MissatgeSignat>...</MissatgeSignat>
</Document>
```

Figura 6-5. Document XML de resposta de consulta d'historial

6.2.6 Document XML de petició del llistat de pacients

El document *XML* de petició del llistat de pacients contindrà el número aleatori generat pel gestor per a autenticar de manera íntegra a l'usuari, en aquest cas només podrà ser un metge, i el DNI de l'usuari corresponent.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
   <Missatge>
      <NumeroAleatoriGestor>
            QGRgwBpDh/7TbYbIsUD52jogqV2X3vbHXbT1zWfChjY=
      </NumeroAleatoriGestor>
      <UsuariDNI>33333333-A</UsuariDNI>
   </Missatge>
</Document>
```

Figura 6-6. Document XML de petició del llistat de pacients

6.2.7 Document XML de resposta del llistat de pacients

El document en format *XML* de resposta que el gestor genera, contindrà el llistat de pacients corresponents a la sol·licitud de l'usuari. Addicionalment, s'afegirà també la signatura del gestor en el document.

A continuació, es pot veure un exemple del contingut d'aquest document.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
      <Missatge>
            <LlistaPacients>
                  <Pacient>
                        <DNI>77777777-A</DNI>
                        <Nom>Joan</Nom>
                        <Cognoms>Poma</Cognoms>
                  </Pacient>

                  <Pacient>
                        <DNI>66666666-A</DNI>
                        <Nom>Laura</Nom>
                        <Cognoms>Girona</Cognoms>
                  </Pacient>
                  <Pacient>
                        <DNI>22222222-A</DNI>
                        <Nom>Sergi</Nom>
                        <Cognoms>Perez</Cognoms>
                  </Pacient>
            </LlistaPacients>
      </Missatge>
      <MissatgeSignat>
            MIIKxAIBATELMAkGBSsOAwIaBQAwCwYJKoZIhvcNAQcBoIIJUzCCBJgwgg
            OAoAMCAQICCQDIsNj2X3GIdjANBgkqhkiG9w0BAQUFADCBlzELMAkGA1UE
            BhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1UEBxMJQmFyY2Vsb2
            5hMQwwCgYDVQQKEwNV...
      </MissatgeSignat>
</Document>
```

Figura 6-7. Document XML de resposta del llistat de pacients

6.2.8 Document XML d'inserció per una visita

Aquest document contindrà, en primer lloc, el número aleatori NG del gestor per a completar l'autenticació del metge. També, s'observa que conté de manera estructurada la informació relativa a la visita juntament amb la corresponent signatura del metge.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
      <Missatge>
            <NumeroAleatoriGestor>
                  S/vuxkLHlsiWrE8TMW68C/O5mpjGaMvWxeyUAKABfxE=
            </NumeroAleatoriGestor>
            <Visita>
                  <Metge>33333333-A</Metge>
                  <Pacient>66666666-A</Pacient>
                  <Diagnostic>Grip</Diagnostic>
                  <Tractament>
                        Dafalgan 1 gram.- 3 cops al dia (durant una
                  set.)
                  </Tractament>
                  <Evolucio>-</Evolucio>
                  <Privat>0</Privat>
            </Visita>
            <SignaturaMetge>
                  MIIKxwIBATELMAkGBSsOAwIaBQAwCwYJKoZIhvcNAQcBoIIJVjCCB
                  JswggODoAMCAQICCQDIsNj2X3GIeDANBgkqhkiG9w0BAQUFADCBlz
                  ELMAkGA1UEBhMCRVMxEjAQBgNVBAgTCUJhcmNlbG9uYTESMBAGA1U
                  EBxMJQmFyY2Vsb25hMQwwCgYDVQQKEwNV...
            </SignaturaMetge>
      </Missatge>
</Document>
```

Figura 6-8. Document XML d'inserció per una visita

6.2.9 Document XML de finalització de sessió

El document *XML* corresponent a la finalització de sessió conté les dades referents al número aleatori NG del gestor i el DNI de l'usuari determinat.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document>
   <Missatge>
      <NumeroAleatoriGestor>
            QGRgwBpDh/7TbYbIsUD52jogqV2X3vbHXbT1zWfChjY=
      </NumeroAleatoriGestor>
      <UsuariDNI>33333333-A</UsuariDNI>
   </Missatge>
</Document>
```

Figura 6-9. Document XML de finalització de sessió

6.3. Diagrama de classes

Per a l'intercanvi de missatges entre els clients (pacients i metges) i el gestor, s'ha fet ús de la tecnologia *XML* per a poder donar un format estàndard en quant a contingut d'aquests missatges que s'intercanvien durant el procés de comunicació entre els diferents entitats que formen el sistema. Per aconseguir donar el format dels documents anteriorment explicats, s'ha implementat una classe encarregada de proporcionar els mètodes i les funcions necessàries per a reproduir aquestes estructures.

En la següent figura, es representa el diagrama de classes (simplificat) en el qual apareix la classe *XML*.

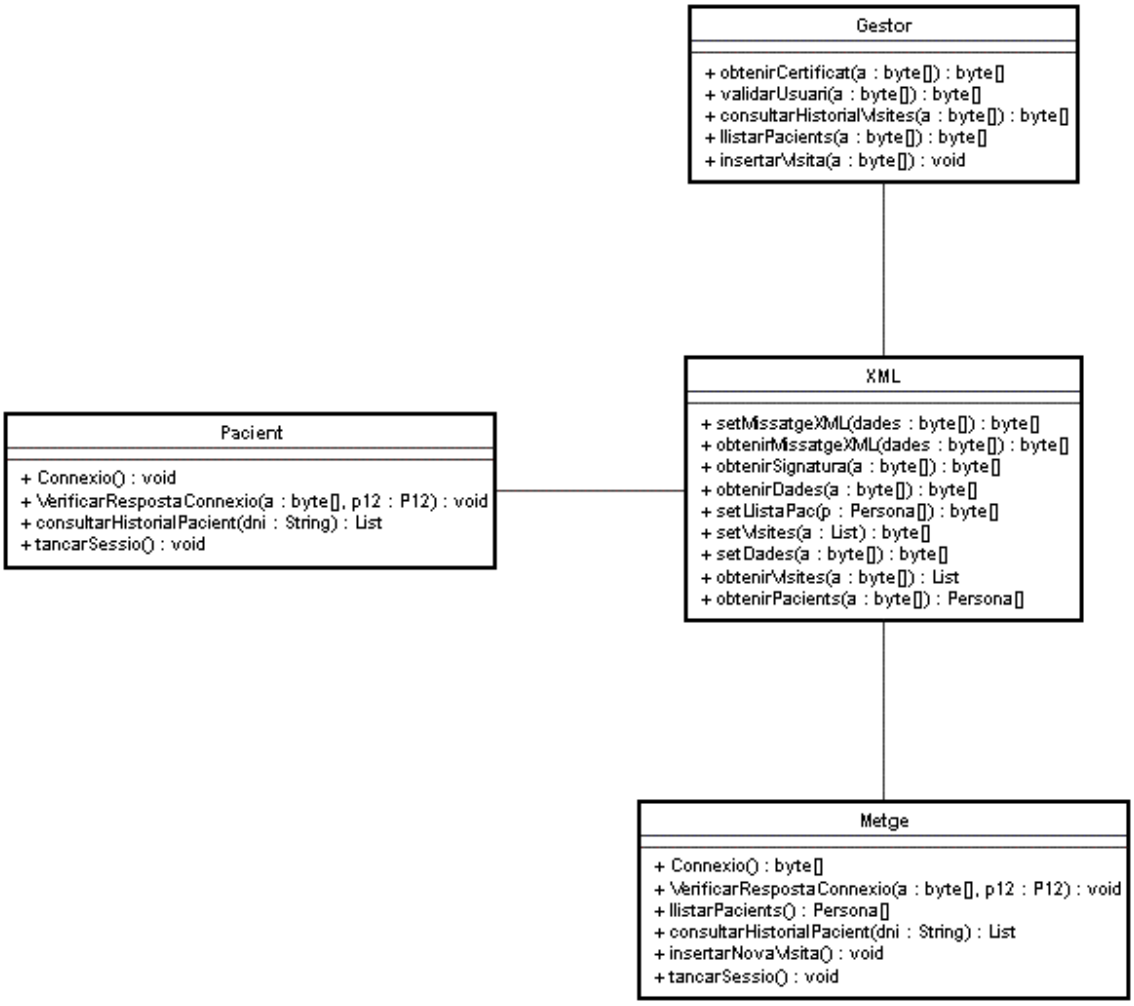

Figura 6-10. Diagrama de classes amb l'XML

Les classes Metge, Pacient i Gestor, han de poder interaccionar amb la classe XML per a confeccionar els missatges que s'intercanviïn i que aquests segueixin el format anteriorment

descrit. D'aquesta manera, aquests obtindran un missatge estructurat en funció de les dades que passin com a paràmetres i del mètode cridat corresponent.

7. RMI

7.1. Introducció

La comunicació dels diferents components és una part essencial d'aquest projecte. Per a aconseguir-ho, aquesta comunicació es realitza mitjançant la invocació de crides remotes. En aquest apartat s'expliquen els conceptes *RMI* més rellevants i com aquests s'apliquen al disseny del sistema de gestió d'historials mèdics.

Els metges han de poder consultar els historials mèdics, visualitzar el llistat dels pacients que tenen assignats i inserir noves visites amb les dades mèdiques corresponents. D'altra banda, el sistema ha de permetre als pacients poder accedir a les dades contingudes al seu expedient mèdic. La invocació dels mètodes que permeten dur a terme aquestes accions, es considera que es pugui efectuar des de la perspectiva d'una vessant remota on les aplicacions client es puguin executar des de localitzacions distants envers la ubicació física del servidor.

El mecanisme d'invocació remota de Java (*RMI*) permet crear aplicacions distribuïdes basades en *Java* en les quals els mètodes remots de *Java* poden ser invocats des de màquines virtuals de *Java* diferents, ja sigui al mateix o diferent ordinador.

7.2. Conceptes generals RMI

Els programes *RMI* engloben una aplicació servidora, que fa accessibles un conjunt d'objectes remots i espera que els clients cridin aquests mètodes (o objectes) remots, i una aplicació client, la qual obté la referència remota d'un o més objectes remots en el servidor i crida als seus mètodes. *RMI* proporciona el mecanisme amb el que es comuniquen i es transmet la informació des del client al servidor o des del client al servidor.

Aplicació genèrica

Així, es poden distingir dos principals components dins de les aplicacions *RMI* i que, els quals, es resumeixen tot seguit:

- Aplicació Servidor:
    - Crea l'objecte remot
    - Crea la referència de l'objecte remot
    - Espera a que un client invoqui un mètode a l'objecte remot
- Aplicació Client:
    - Obté una referència a un objecte remot en el servidor
    - Invoca un mètode remot

Funcionalitats

Aquest tipus d'aplicacions d'objectes distribuïts requereixen:

- localitzar els objectes remots
    - en *RMI* s'implementa un registre (*rmiregitry*) que actua com a servidor de noms per a permetre la localització dels objectes remots.
- comunicar-se amb els objectes remots
    - es realitza amb el propi sistema *RMI*, essent transparent per al programador.
- Pas de paràmetres i de resultats en els mètodes remots
    - inclou el tractament de referències i de la càrrega dinàmica de classes.

Arquitectura

El sistema *RMI* està format per una arquitectura en quatre capes en que cada una té una funció específica.

<table>
<tr><td>Aplicació Client</td><td rowspan="2">Capa d'aplicació</td><td>Aplicació Servidor</td></tr>
<tr><td>Stub</td><td rowspan="2">Capa de representant (proxy)</td><td>Esquelet (skeleton)</td></tr>
<tr><td>Referència Remota</td><td rowspan="2">Capa de RMI</td><td>Referència Remota</td></tr>
<tr><td colspan="3">Transport</td></tr>
</table>

Figura 7-1. Arquitectura RMI

La capa d'aplicació és on es troben els objectes que implementen l'aplicació. D'una banda, es troben les aplicacions clients, que són els que invoquen els mètodes o emeten peticions a d'altres objectes remots. Per l'altra, s'identifiquen els objectes servidors, els quals reben peticions d'altres objectes remots.

La capa de representant és la que inclou els objectes que actuen com a representants locals dels objectes remots. S'encarreguen d'empaquetar i desempaquetar (*marshalling*) les invocacions considerant els arguments i els resultats dels mètodes remots. Existeixen dos tipus de representants: els *stubs* (del costat client) i els esquelets (del costat del servidor).

En la capa de referència remota es realitza la interpretació de les referències a objectes remots, les referències locals a *stubs* o *skeleton* es resolen amb les seves parts remotes respectives i aquestes dades, juntament amb els paquets (*marshalled*) que contenen les peticions o els resultats de les invocacions, es passen a la capa de transport.

La capa de transport és la que es troba en el protocol de comunicació (*TCP/IP*), i s'encarrega de transportar els missatges que intercanvien diferents objectes.

7.3. Diagrama de classes

En la Figura 7-2 es mostra una ampliació del diagrama de classes on queden modificades les relacions entre Pacient i Metge amb la classe Gestor. Ara, apareix una classe entremig anomenada InterficieRemota, encarregada de definir i referenciar l'objecte remot (la classe Gestor) i els mètodes que implementa.

Així, havent adaptat el sistema per a poder rebre invocacions remotes, les classes que intervenen en aquesta modificació són les següents:

- InterficieRemota és una interfície *Java* amb la declaració de tots els mètodes que es poden invocar de forma remota, és a dir, els mètodes que es volen cridar des del client però que s'executaran en el servidor.

- La classe Gestor esdevé l'objecte que implementa la interfície remota definida i que només és vist pel servidor *RMI*.

- Servidor s'encarrega d'instanciar l'objecte remot Gestor i de registrar-lo en el *rmiregistry* per a posar aquest objecte a disposició dels clients.

- Pacient i Metge esdevenen les classes client encarregades d'utilitzar l'objecte remot (els seus mètodes) demanant una referència remota.

Tot seguit es mostra una representació simplificada del diagrama de classes adaptat per a suportar les invocacions remotes implementades per *RMI*.

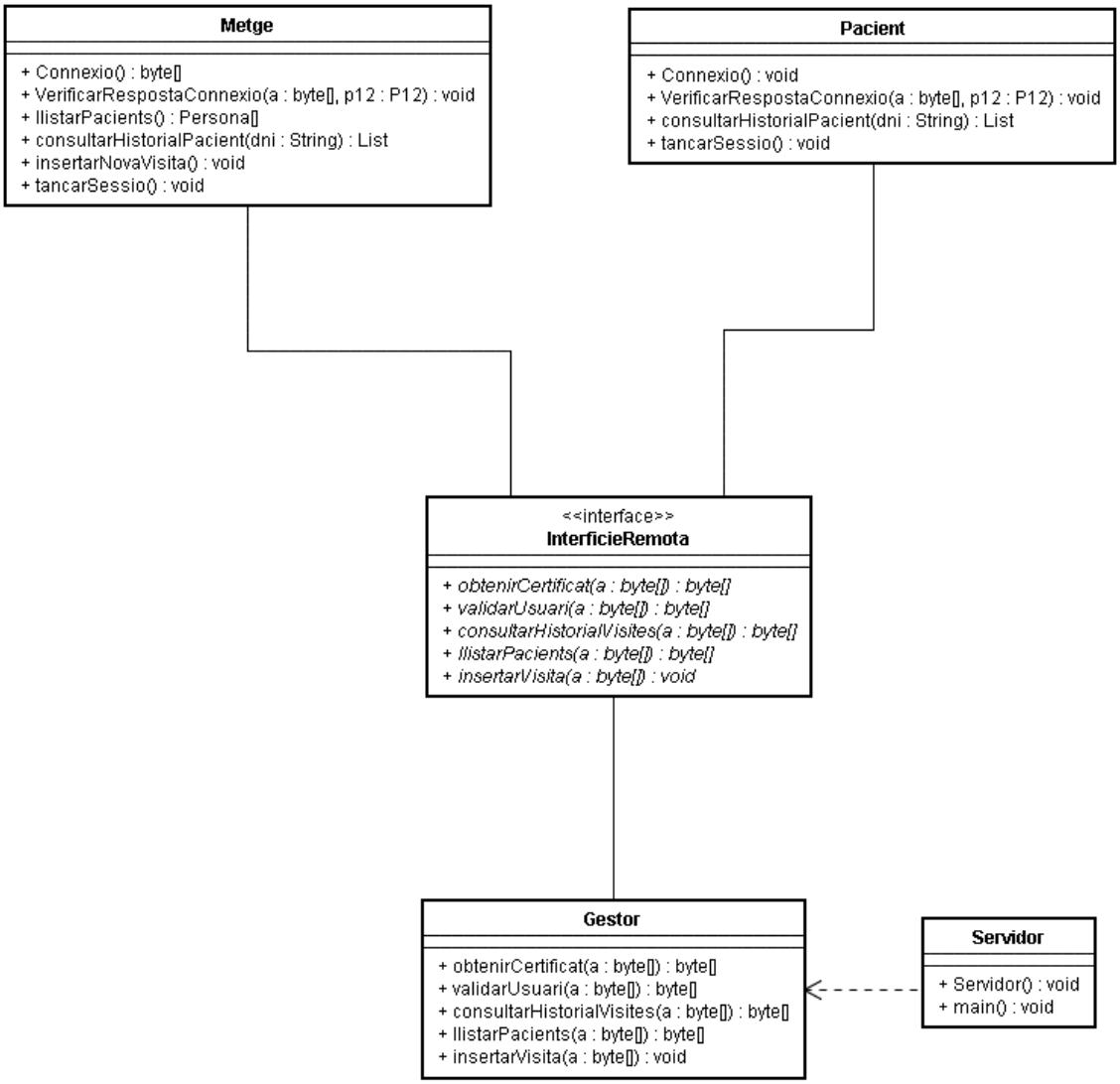

Figura 7-2. Diagrama de classes amb Interfície Remota per RMI

8. Base de dades

8.1. Introducció

La utilització d'una base de dades permetrà emmagatzemar tota aquella informació que és necessària per al bon funcionament de l'aplicació i del sistema en general. L'administració i gestió de les dades tractades és un dels punts més importats per a aquest projecte.

La gestió de la informació és necessària i dóna lloc a permetre mantenir l'estat persistent sobre els usuaris de l'aplicació i sobre les dades dels historials i visites tractades. La base de dades permetrà que les dades esdevinguin dins d'un rol persistent i, alhora, perdurable en el temps.

Les dades s'han d'organitzar d'acord un procés previ que comprèn l'anàlisi i el model de dades, i l'elecció i posterior configuració del sistema gestor que suportarà la nostra base de dades.

8.2. Implementació de la base de dades

Per a aquest projecte, s'ha optat per l'ús de *MySQL* com a sistema gestor de base de dades relacional. Es pot fer ús de *MySQL* en aplicacions de tota mena i de forma lliure i gratuïta sota les condicions de llicència *GPL* (per àmbit no comercial o acadèmic). Tanmateix, aquest sistema gestor està disponible per a multitud de plataformes. Aquests, esdevenen els motius principals del per què s'ha escollit *MySQL*, com el sistema gestor del sistema.

Encara que segons el disseny del sistema no caldria que la base de dades suportés transaccions, s'ha decidit utilitzar una implementació de la base de dades configurada capaç de proporcionar un entorn transaccional en vista a futures adaptacions que ho pogués requerir. Així, s'han implementat les taules amb el motor d'emmagatzemament anomenat *InnoDB*, proporcionat per *MySQL*, que permet disposar d'un entorn amb suport a transaccions.

D'altra banda, s'ha cregut convenient emmagatzemar els certificats d'usuari corresponents, i la creació de certs camps que faran referència a xifratge i signatures, mantenint un format de tipus binari anomenat *BLOB* (*Binary Large Object*).

Per a l'accés a la base de dades des de la part del servidor de l'aplicació, el gestor del sistema, s'ha utilitzat el connector *JDBC* (*Java DataBase Connectivity*), el qual permet accedir a la base de dades *MySQL* des d'aplicacions desenvolupades en *Java*.

8.3. Model relacional de la base de dades

Tot seguit, es presenta el diagrama referent al model de dades que s'utilitza per a la gestió i manipulació de la informació tractada en el sistema.

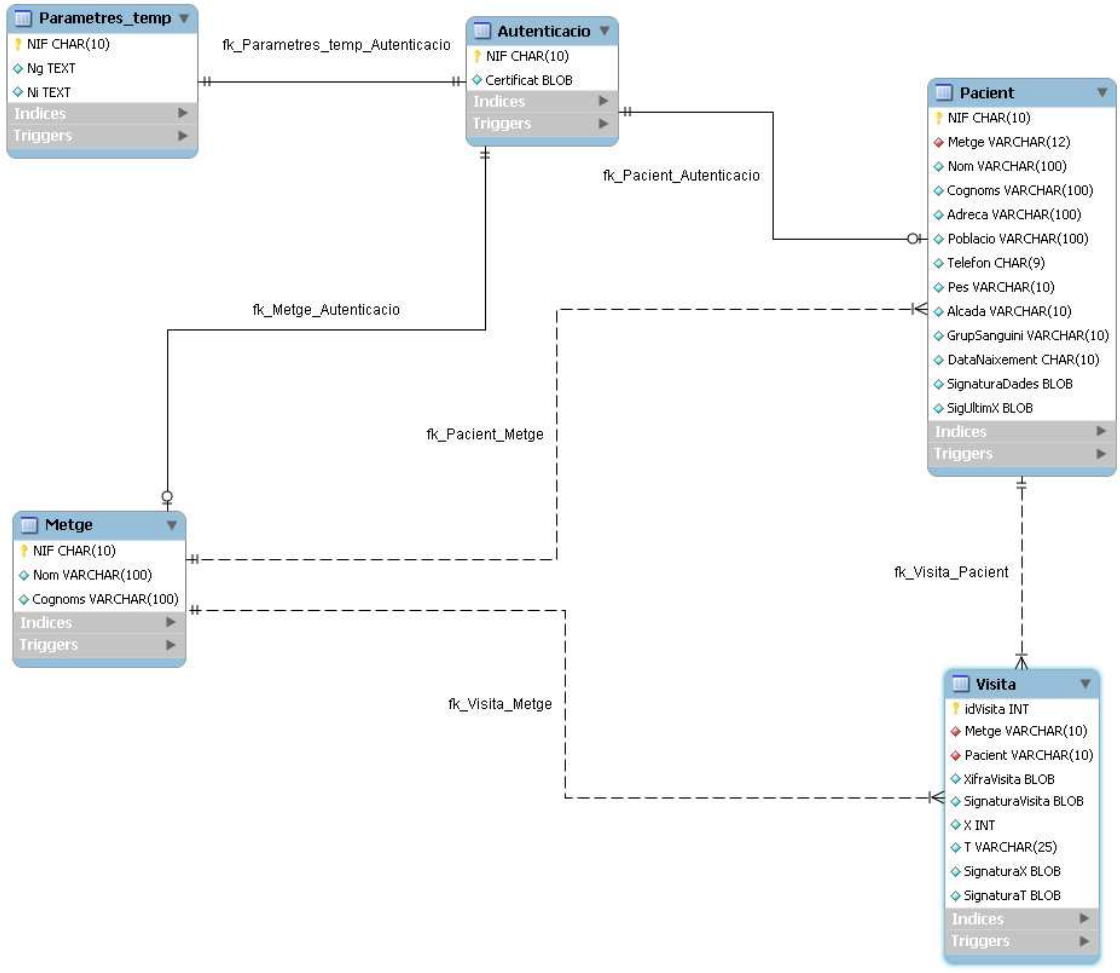

Figura 8-1. Diagrama relacional de la base de dades

En aquest model es poden observar les relacions que es produeixen entre les diferents entitats que conformen la base de dades.

- Parametres_temp → Autenticacio: es tracta d'una relació d'u a u en la qual es vol relacionar els paràmetres temporals d'un determinat usuari que ha estat autenticat amb la taula autenticació que conté el NIF de l'usuari en qüestió juntament amb el seu certificat. Es podria dir que s'estableix una relació de dependència en què la taula autenticació, que és la que conté les dades identificadores més importants de cada usuari, dóna cap a la taula paràmetres temporals. És a dir, paràmetres temporals depèn de la taula autenticació la qual li proporciona, a més a més, una identificació. Tanmateix, però, tot i ser una relació d'u a u en què la taula de paràmetres temporals sempre correspondrà a una autenticació concreta, la primera pot no estar inicialitzada (nul·la).

- Metge → Autenticacio: és tracta d'una relació de zero a u. Un metge sempre tindrà una autenticació, però una autenticació no sempre farà referència a un metge. Aquí, de nou, es podria considerar que existeix una relació de dependència donat que la clau forana de metge és també l'identificador d'aquest.

- Pacient → Autenticacio: es tracta d'una relació de zero a u. Un pacient sempre tindrà una autenticació, però una autenticació no sempre farà referència a un pacient. De la mateixa manera, es pot considerar que existeix una relació de dependència donat que la clau forana de pacient és també l'identificador d'aquest.

- Metge → Pacient: s'estableix una relació d'u a molts, en què un metge pot tenir molts pacients assignats.

- Metge → Visita: es tracta d'una relació d'u a molts entre la taula metge i la taula visita. Un metge pot realitzar moltes visites i, una visita, sempre ha d'estar assignada a un únic metge que és qui la realitza.

- Pacient → Visita: es tracta d'una relació d'u a molts entre la taula pacient i la taula visita. Un pacient pot tenir moltes visites al llarg de cert període de temps i, una visita, sempre farà referència a un únic pacient.

8.4. Descripció de les taules de la base de dades

8.4.1. Autenticacio

Aquesta taula és l'encarregada d'emmagatzemar la informació relativa a l'autenticació dels usuaris. Com a atributs de la taula destaquem:

- NIF: clau primària de la taula i farà referència al NIF d'un pacient o un metge.
- Certificat: camp que emmagatzema el certificat corresponent a l'usuari (metge o pacient).

8.4.2. Parametres_temp

Aquesta taula és l'encarregada de guardar els paràmetres temporals que es produeixen en el moment que l'usuari inicia la sessió en el sistema.

- NIF: esdevé identificador i correspon al NIF de l'usuari que inicia la sessió. Fa referència també a l'autenticació.

- Ng: camp que emmagatzema el número aleatori generat pel gestor.
- Ni: camp que emmagatzema el número aleatori generat per l'usuari.

8.4.3. Pacient

Aquesta taula emmagatzema la informació referent als pacients. Lògicament, es concep la separació de les dades personals (NIF, nom, cognoms, adreça, població i telèfon) i de les dades mèdiques (pes, alçada i grup sanguini) del pacient. Per a facilitar la gestió de les dades i no extendre's excessivament, aquesta informació s'ha decidit incorporar-la junta en aquesta mateixa taula.

També es destaca la conveniència d'incorporar del camp SignaturaDades per a disposar una mesura de seguretat addicional. Aquest camp correspon a la signatura de les dades introduïdes que fan referència al pacient.

Així, doncs, la taula està formada pels següents camps:

- NIF: NIF del pacient i que és considerat com a identificador únic. També és considerat com a clau forana que fa referència a autenticació.
- Metge: clau forana referent al metge al qual el pacient és assignat.
- Nom: nom del pacient.
- Cognoms: cognoms del pacient.
- Adreca: adreça del pacient.
- Poblacio: població del pacient.
- Telefon: telèfon del pacient.
- Pes: pes del pacient.
- Alcada: alçada del pacient.
- GrupSanguini: grup sanguini del pacient.
- DataNaixement: data de naixement del pacient.
- SignaturaDades: signatura corresponent a les dades del pacient.
- SigUltimX: signatura corresponent al darrer número de visita referent al pacient.

8.4.4. Metge

Aquesta taula emmagatzema les dades dels metges existents en el sistema. Està composta dels següents camps:

- NIF: NIF del metge i que és considerat com a identificador únic. També és considerat com a clau forana que fa referència a autenticació.
- Nom: nom del metge.
- Cognoms: cognoms del metge

8.4.5. Visita

Aquesta taula és l'encarregada d'emmagatzemar les visites efectuades per un metge determinat per a un pacient concret. La taula està formada pels següents camps:

- idVisita: clau primària de la taula.
- Metge: clau forana que identifica el metge.
- Pacient: clau forana que identifica al pacient.
- XifraVisita: dades de la visita xifrades.

- SignaturaVisita: signatura de la visita.
- X: número de sèrie X.
- T: instant de temps T.
- SignaturaX: signatura del número de sèrie X.
- SignaturaT: signatura de l'instant de temps T.

8.5. Diagrama de classes

Tal i com s'ha comentat, la base dades contindrà tota aquella informació necessària i que cal emmagatzemar en relació a les sessions d'usuari, certificats, metges, pacients i les visites dels pacients. Es destaca que només el gestor tindrà accés a la base de dades, i és per això, que les dades que siguin demanades pels clients, el gestor serà l'encarregat de, primer, cercar la informació demanada i, segon, proporcionar-la responent a la sol·licitud.

Per a què el gestor pugui accedir a la base de dades, s'ha implementat una classe anomenada BD on s'inclouen els mètodes i funcions necessàries per a dur a terme les operacions de consulta, inserció, actualització i eliminació de les dades.

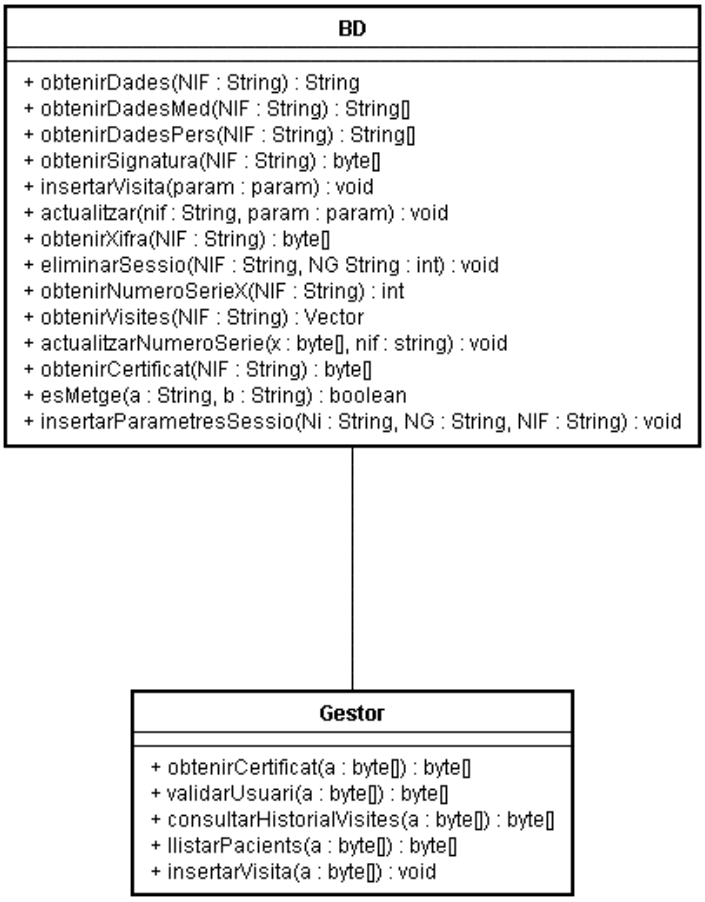

Figura 8-2. Diagrama de classes relacionades per a l'accés a la BD

En la figura anterior, s'ha representat la nova classe que dóna la funcionalitat per a la gestió de la informació, és a dir, accés a la base de dades. Aquesta nova classe s'incorpora, doncs, al diagrama de classes general.

No s'ha necessitat instal·lar cap programari addicional, sinó que s'ha fet servir *JDBC* proporcionat per l'*API* estàndard de *Java* per a programar els mètodes que gestionen l'accés a la informació.

9. Interfície gràfica

9.1. Introducció

L'objectiu de la interfície gràfica és afavorir i facilitar la interacció dels usuaris, pacients i metges en el cas d'aquest projecte, per a poder executar les funcionalitats i dur a terme les accions per a les quals el sistema es concebut.

D'aquesta manera, els components que formen la interfície permetran provar les accions sobre els usos descoberts en els requeriments inicials i identificats en el diagrama de casos d'ús del sistema i que els diferents actors duen a terme.

Tot i la simplicitat de la interfície gràfica que s'ha dissenyat, aquesta, manté una forma adequada d'acord en satisfer unes necessitats, que es podrien considerar mínimes, i permetre obtenir una visió clara i concreta de com dur a terme les diferents funcions.

En aquest capítol es poden veure una representació de les pantalles principals de l'aplicació metge i de l'aplicació pacient, i com aquestes segueixen un determinat estil.

9.2. Implementació de la interfície gràfica

Per a la implementació de la interfície gràfica, s'ha utilitzat un editor visual que es pot incorporar des de les darreres versions de l'*Eclipse*, el qual permet disposar d'un entorn de creació i disseny, fent ús d'alguna llibreria gràfica.

En la confecció de les pantalles s'ha decidit fer ús de la biblioteca gràfica *Swing* per a *Java* que proporciona accés de forma senzilla a tot un ventall ampli de components gràfics.

Tal i com es s'ha definit en els objectius del projecte, s'ha dissenyat un aplicatiu per als metges i, un altre, per als pacients. Aquest fet té en compte que els metges disposen de més accions per a executar que els metges. S'ha considerat que, aquesta, era una bona opció per separar els

components gràfics necessaris per a dur a terme les accions, anteriorment estudiades, en funció de l'usuari que hagi de tenir accés al sistema.

Cal destacar que, malgrat diferenciar ambdós aplicatius, tots dos presenten una visualització similar, canviant només les accions que l'usuari pot dur a terme en relació al seu rol.

9.3. Aplicació Metge

9.3.1. Pantalla principal

La primera pantalla que apareix en el moment d'executar l'aplicació és la finestra principal o de benvinguda i en la que es mostraran a dins la resta de finestres de l'aplicatiu. Aquesta finestra presenta un menú, situat en la barra superior, on l'usuari pot començar a interaccionar amb l'aplicació. Es destaquen tres possibles events: Arxiu, Opcions i Ajuda. En la primera, només s'inclou l'opció de sortir. En la segona, s'inclouen les opcions d'autenticar-se (apareixerà activa), de llistar pacients, consulta d'historials i finalització de sessió (aquestes tres darreres inactives). Per últim, es pot observar una darrera possibilitat, l'Ajuda (tan sols inclou la possibilitat de veure una petita referència anomenada "sobre l'aplicació".

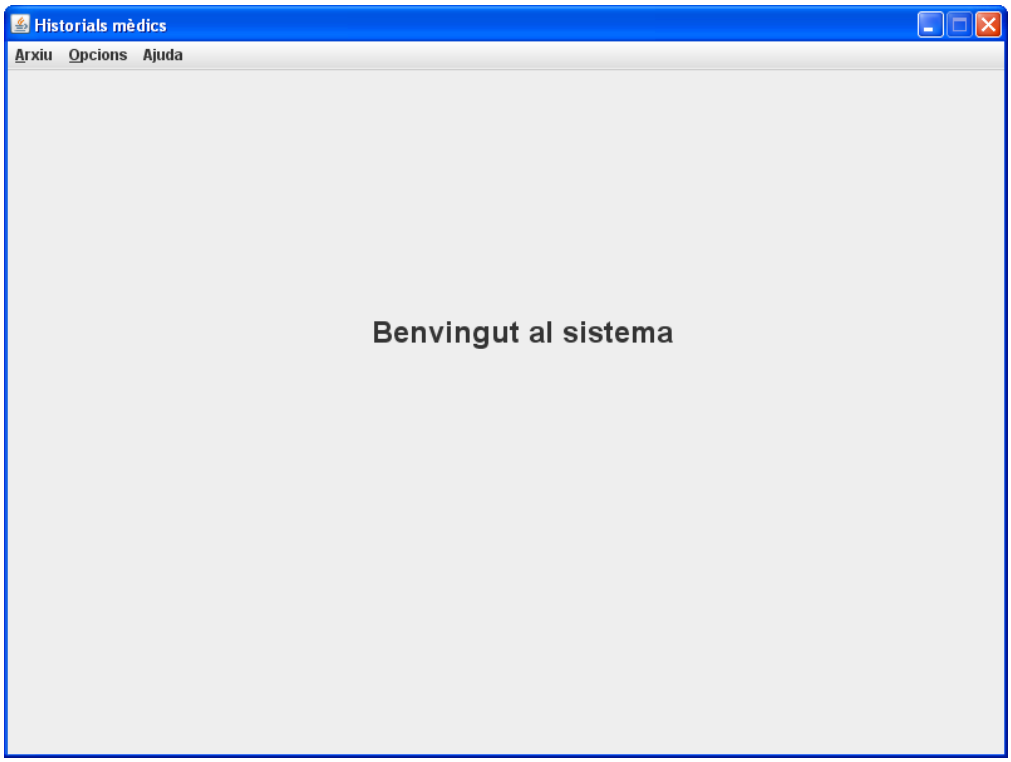

Figura 9-1. Pantalla principal

Un cop es despleguen les Opcions establertes en el menú superior, l'usuari pot veure quines opcions pot executar. Inicialment, només pot dur a terme la funció d'autenticació al sistema, mentre que les altres (llistar pacients, consulta d'historials i finalització de sessió) es presenten com inactives fins que l'usuari no hagi procedit, i dut a terme correctament, el procés d'autenticació.

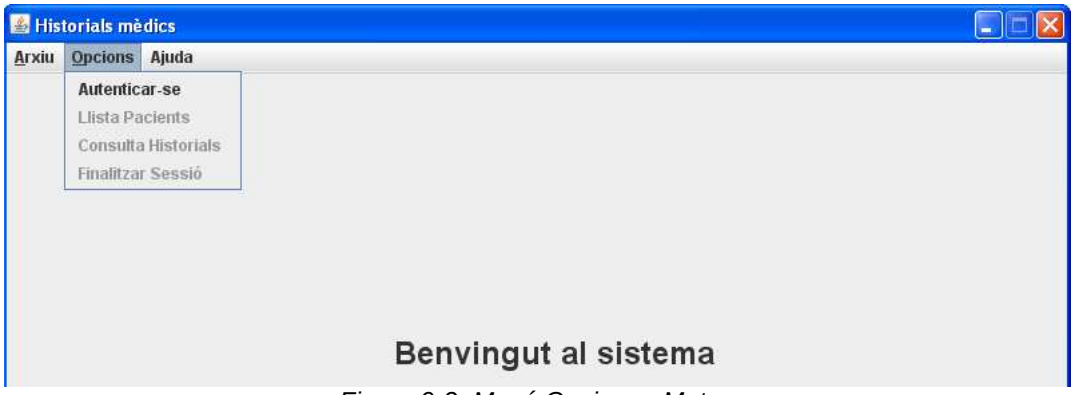

Figura 9-2. Menú Opcions - Metge

9.3.2. Pantalla d'autenticació

Un cop l'usuari ha triat l'Opció d'autenticació, es mostrarà una finestra on aquest pot dur a terme l'acció.

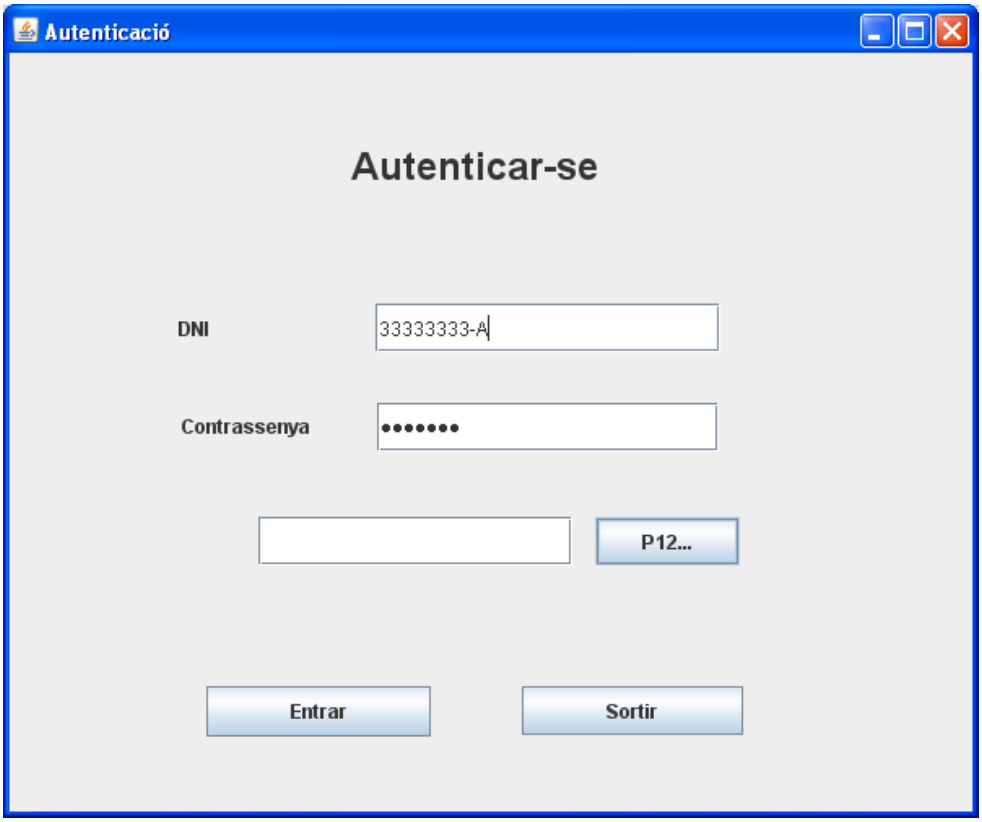

Figura 9-3. Pantalla d'autenticació

En aquesta pantalla es poden observar els dos camps que l'usuari haurà d'omplir, el DNI i la contrasenya. També, s'ha inclòs un botó per a facilitar que l'usuari triï la ruta de l'arxiu P12 corresponent. Un cop l'usuari premi el botó P12, apareixerà de forma emergent un component

visual on aquest podrà especificar, navegant per l'estructura de directoris i carpetes del seu sistema, el seu arxiu P12.

Una vegada omplerts tots els camps, l'usuari pot emetre l'acció per entrar i autenticar-se amb les dades introduïdes, envers al sistema.

En el moment que l'usuari ha estat autenticat, la finestra principal canviarà indicant que s'ha establert la connexió i, mostrant les opcions del menú superior actives, a excepció de la autenticació (procés ja realitzat i actiu).

Figura 9-4. Connexió establerta - Metge

Ara, l'usuari podrà triar entre una de les Opcions del menú situat en la part superior. A continuació, es dóna una mostra de les diferents pantalles que apareixen en funció de l'Opció que sigui triada per l'usuari.

9.3.3. Pantalla Llista Pacients

Un cop ha triat l'Opció per llistar els pacients, apareixerà la finestra corresponent amb els pacients assignats al metge autenticat. En aquesta pantalla es podrà seleccionar una fila que correspongui a un dels pacients de la llista en la qual apareixen, i poder consultar l'historial del pacient que ha estat seleccionat.

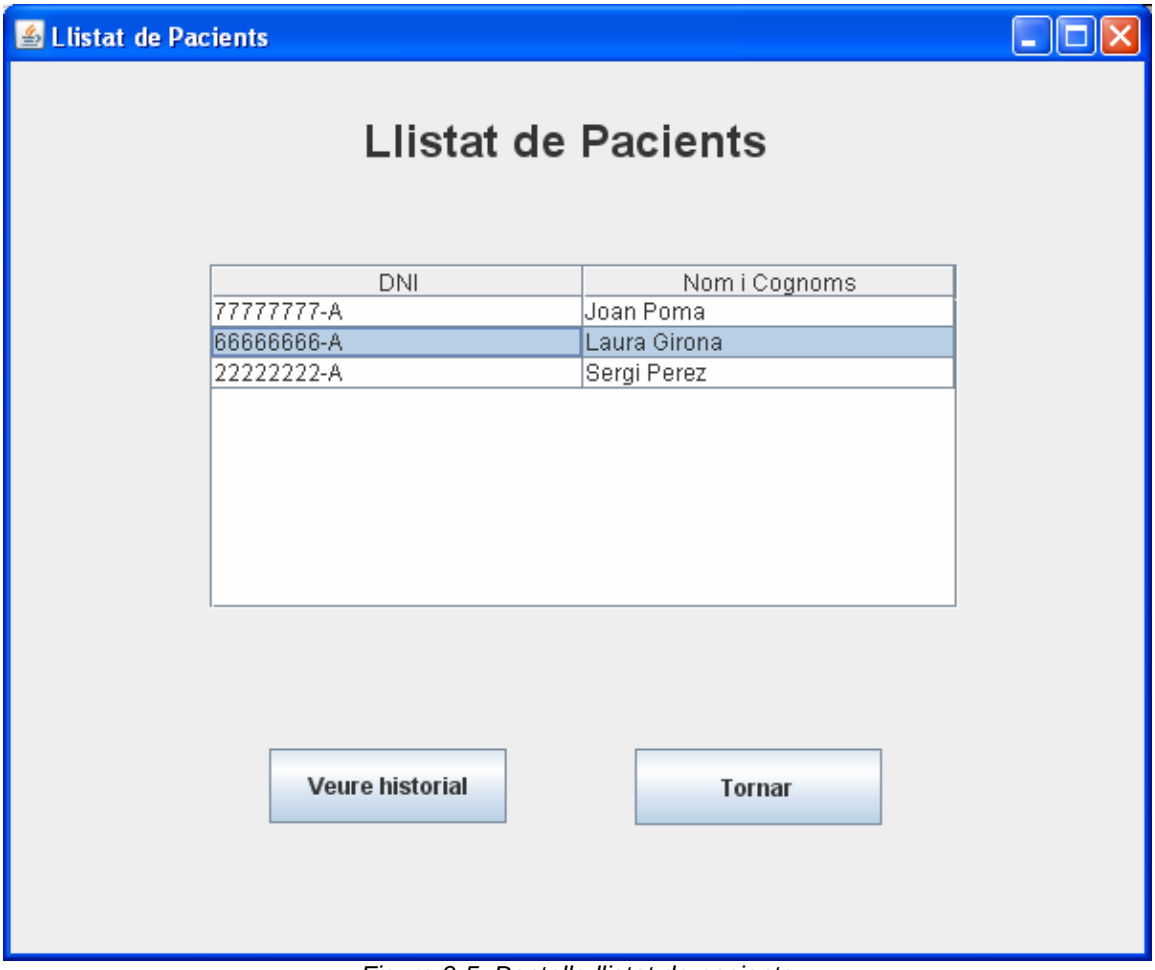

Figura 9-5. Pantalla llistat de pacients

9.3.4. Pantalla Historial del pacient

Si el metge ha seleccionat un dels pacients de la llista de la pantalla on es llistaven els pacient i premut el botó "Veure historial", o, si ha dut a terme la pantalla Consulta Historial per DNI (veure secció 9.3.6. d'aquest mateix capítol), apareixerà una nova finestra on visualitzarà l'historial del pacient seleccionat (o amb el DNI que s'hagi introduït en la pantalla Consulta Historial per DNI).

En aquesta pantalla es poden observar diferents pestanyes en les que es fa referència a la informació relativa a l'historial mèdic del pacient seleccionat.

- Pestanya Dades Personals: el metge podrà consultar les dades personals del pacient: NIF, Nom, Cognoms, Adreça, Població, Telèfon i Data de Naixement.

- Pestanya Dades Mèdiques: del metge podrà consultar les dades mèdiques del pacient i que fan referència a: Pes, Alçada i Grup Sanguini.

- Pestanya Historial: si el metge ho desitja, podrà tenir accés a les dades corresponents a les visites sobre l'historial del pacient. Tot aquest conjunt d'informació és el que formaria l'expedient del pacient.

Seguidament, es pot veure un exemple de la pantalla Historial del Pacient i de com aquesta està formada.

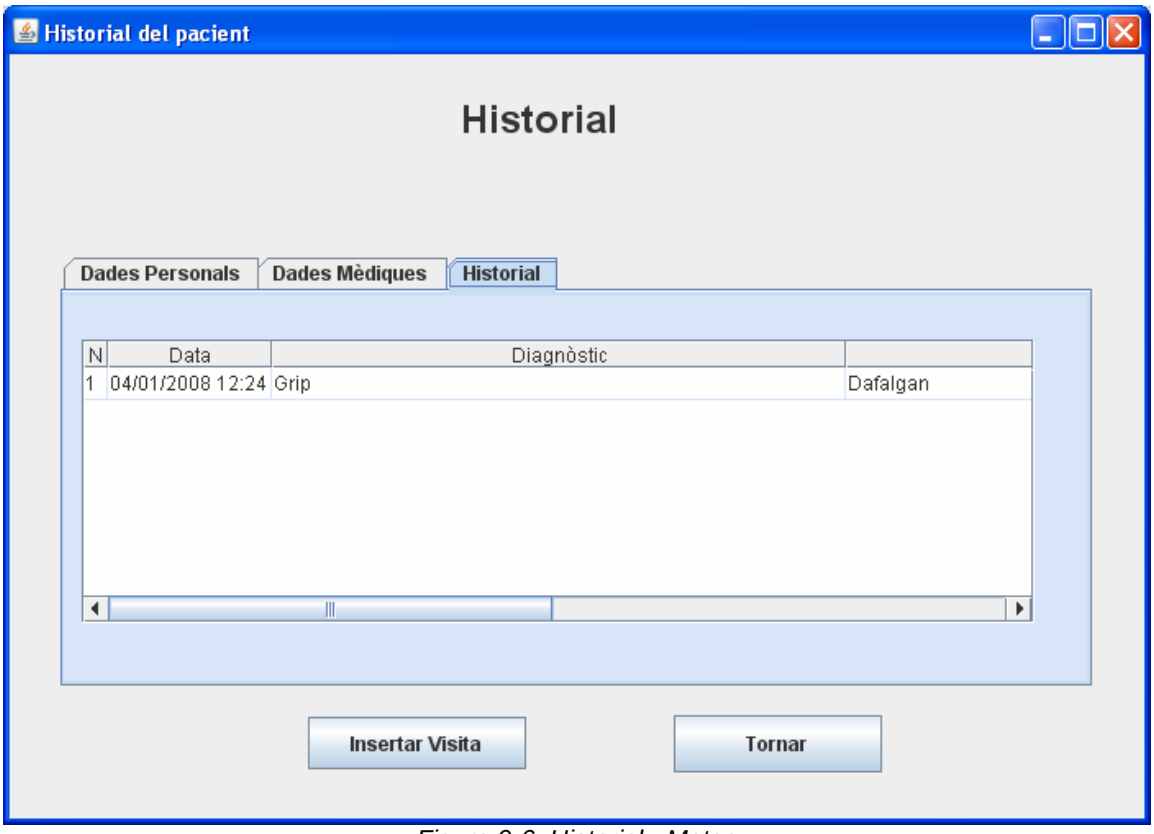

Figura 9-6. Historial - Metge

Seguint fent referència a la pestanya que conté les dades de l'historial es pot veure com les dades de la visita es composen per: N (número de la visita corresponent), Data (instant en què es va donar la visita), Diagnòstic (decisió de què li passava al pacient en funció dels símptomes presentats) Tractament (medicació o acció presa pel metge per a contrarestar la causa) i Evolució (en cas que existeixi algun tipus de procés de llarga durada i que requereixi fer un seguiment en quant a visites).

Tot i que no es pugui apreciar del tot, en aquesta finestra existeix un *scroll* horitzontal per a poder visualitzar bé els camps que composen l'historial del pacient.

En aquesta mateixa pantalla el metge té l'opció de prémer el botó que li comunicarà amb una nova finestra per a poder dur a terme la inserció d'una nova visita. Cal destacar, que si s'accedeix a l'historial del pacient a través de la pantalla anomenada Consulta Historial per DNI, aquest botó queda automàticament desactivat, és a dir, tan sols es podrà accedir a la inserció de la visita a través d'haver seleccionat un pacient de la pantalla Llista de pacients.

9.3.5. Pantalla Inserir Visita

En la propera pantalla es pot veure una mostra de com és el disseny de la pantalla d'inserció de visites que el metge hi pot accedir. Aquesta pantalla està formada pels camps de text a omplir següents:

- Diagnòstic
- Tractament
- Evolució

Tanmateix, s'ha inclòs un *checkbox* per a que el metge pugui indicar si una visita hauria de tenir un tractament privat.

Figura 9-7. Insertar Visita

Finalment, destacar els dos botons d'aquesta finestra:

- Insertar: cridarà l'esdeveniment per a inserir la visita amb les dades que s'han omplert en els camps de text.
- Tornar: situarà al metge a la finestra de menú principal.

9.3.6. Pantalla Consulta historial per DNI

Tal i com s'ha destacat en la secció 9.3.4, el metge té la possibilitat d'accedir a la consulta d'un historial mèdic introduint el DNI d'un pacient. Quan s'accedeix a la pantalla Historial del pacient a través de la pantalla Consulta historial per DNI, queda desactivat el botó Insertar Visita (veure representació de la pantalla en la secció 9.3.4). Aquesta és una decisió de disseny que s'ha considerat per comoditat donat que els metges poden consultar les dades contingudes als historials mèdics de qualsevol pacient (a excepció de les dades privades, en què tan sols les podrà visualitzar en cas de que sigui un pacient assignat al metge en qüestió), però només podran incorporar una nova visita en cas de que sigui un pacient assignat al metge. Per a poder inserir una visita des de la pantalla d'historials, doncs, cal haver accedit prèviament de la pantalla referent al llistat dels pacients assignats.

Figura 9-8. Pantalla consultar per DNI

El botó Consultar executarà la sol·licitud de consulta de l'historial a partir del DNI del pacient introduït en el camp de text que es visualitza en la pantalla. El botó Tornar, retornarà a l'usuari a la pantalla principal de l'aplicació.

9.4. Aplicació Pacient

9.4.1. Pantalla Principal

L'aplicatiu del pacient és similar a l'aplicació que forma la interfície gràfica per als metges. Només presenten algunes diferències que es detallen a continuació.

Respecte a la pantalla principal, en l'aplicació pacient també es mostra el mateix tipus de menú de la barra superior que conté: Arxiu (amb opció de Sortir de l'aplicació), Opcions (només podrà autenticar-se, consultar l'historial mèdic i finalitzar sessió) i Ajuda (amb una finestra que mostrarà una petita referència anomenada "Sobre l'aplicació").

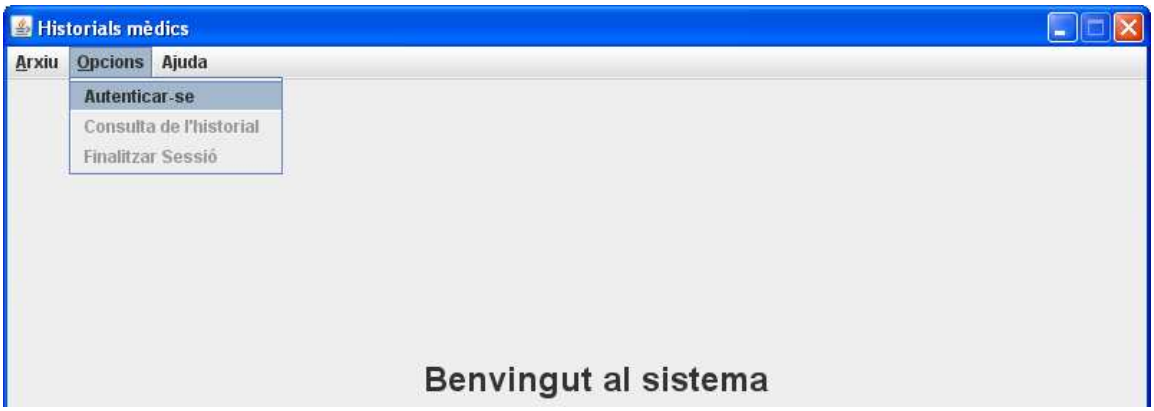

Figura 9-9. Menú Opcions - Pacient

Així, doncs, com a principal diferència a destacar és que existeix un nombre més reduït d'Opcions a executar. L'usuari amb rol pacient, només podrà dur a terme les opcions d'autenticació, de consulta del seu propi historial mèdic i per a poder finalitzar la sessió (si ja ha estat autenticat amb anterioritat).

La pantalla d'autenticació del pacient, presenta la mateixa estructura i disseny a l'anteriorment descrita en la secció 9.3.2. Pantalla d'autenticació (veure la secció esmentada, en aquest mateix capítol, per a obtenir una referència visual de la pantalla).

Una vegada el pacient s'ha autenticat enfront del sistema, les Opcions del menú superior, quedaran de la manera que segueix a continuació.

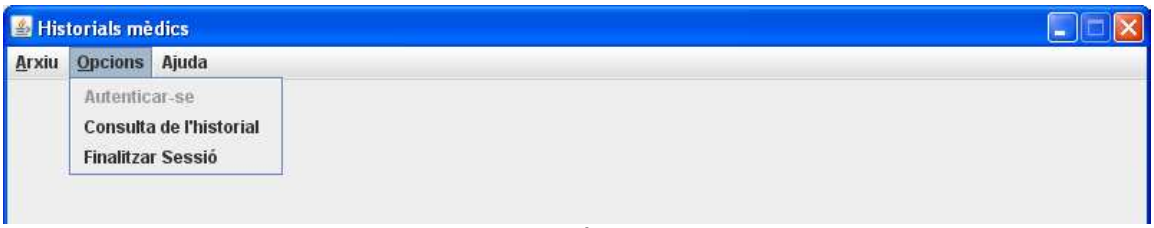

Figura 9-10. Connexió establerta - Pacient

El pacient només tindrà l'opció de consultar el seu propi historial mèdic, o bé finalitzar la sessió actual.

9.4.2. Pantalla Consulta de l'historial

Aquesta finestra mante la mateixa estructura i disseny que la que s'ha descrit anteriorment en la secció 9.3.4. amb l'única diferència de que al pacient no se li és permès realitzar insercions al seu historial. És per aquesta raó, que el botó Inserir Visites (veure secció 9.3.4. d'aquest mateix capítol) no existeix en aquesta finestra. En aquest cas, el pacient només podrà prémer el botó Tancar per retornar al menú principal.

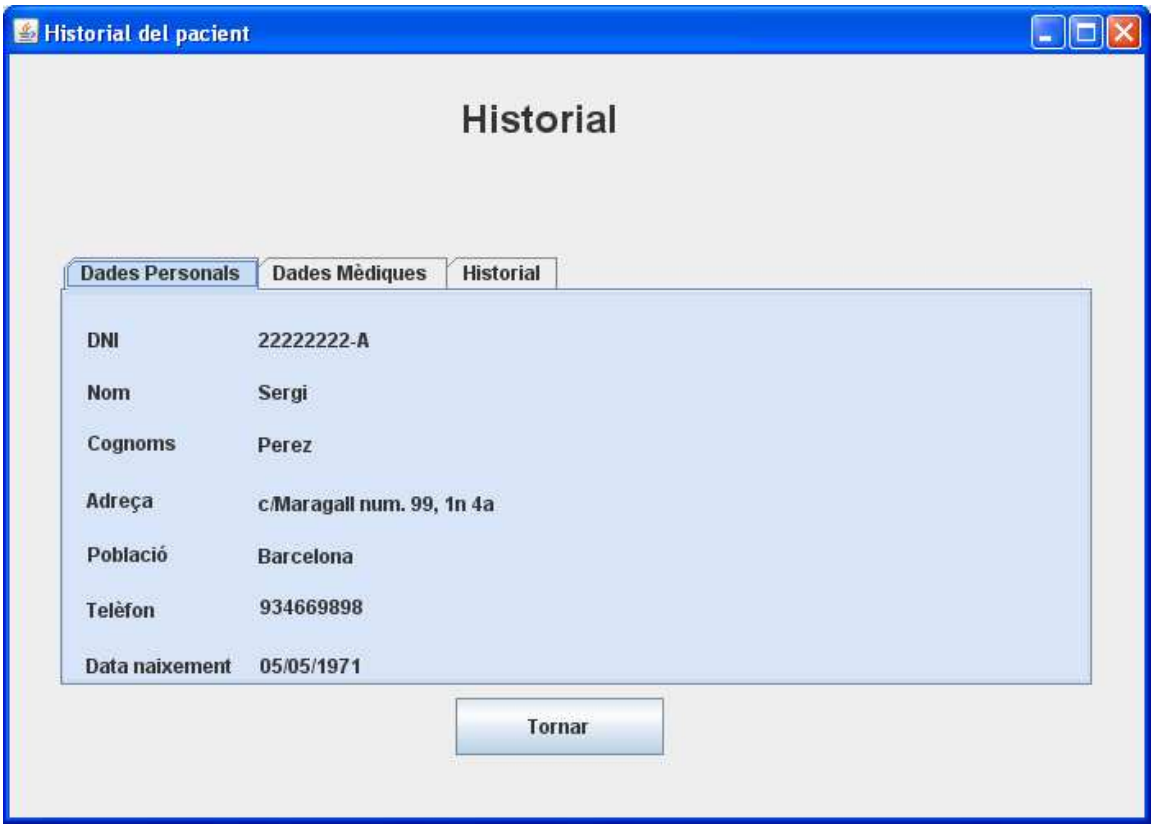

Figura 9-11. Historial - Pacient

9.5. Diagrama de classes

En la implementació de la interfície gràfica s'han separat les classes que formen les vistes en funció del tipus d'usuari. Així, si l'usuari es tracta d'un metge, executaria la seva aplicació a partir de les classes que la composen. De la mateixa manera, si es tracta d'un pacient, aquest, executarà la seva aplicació corresponent.

S'ha pres la decisió de distingir la implementació de les vistes en funció del tipus d'actor que les executa donat que cadascun projecta unes necessitats de serveis i d'ús diferents (veure capítols 2.3 i 2.4).

9.5.1. Aplicació Metge

Per a la confecció de la interfície gràfica de l'aplicatiu de la part referent al metge, s'han fet servir diverses noves classes que podríem anomenar vistes**.** Seguidament, es representa el diagrama de classes que es defineix.

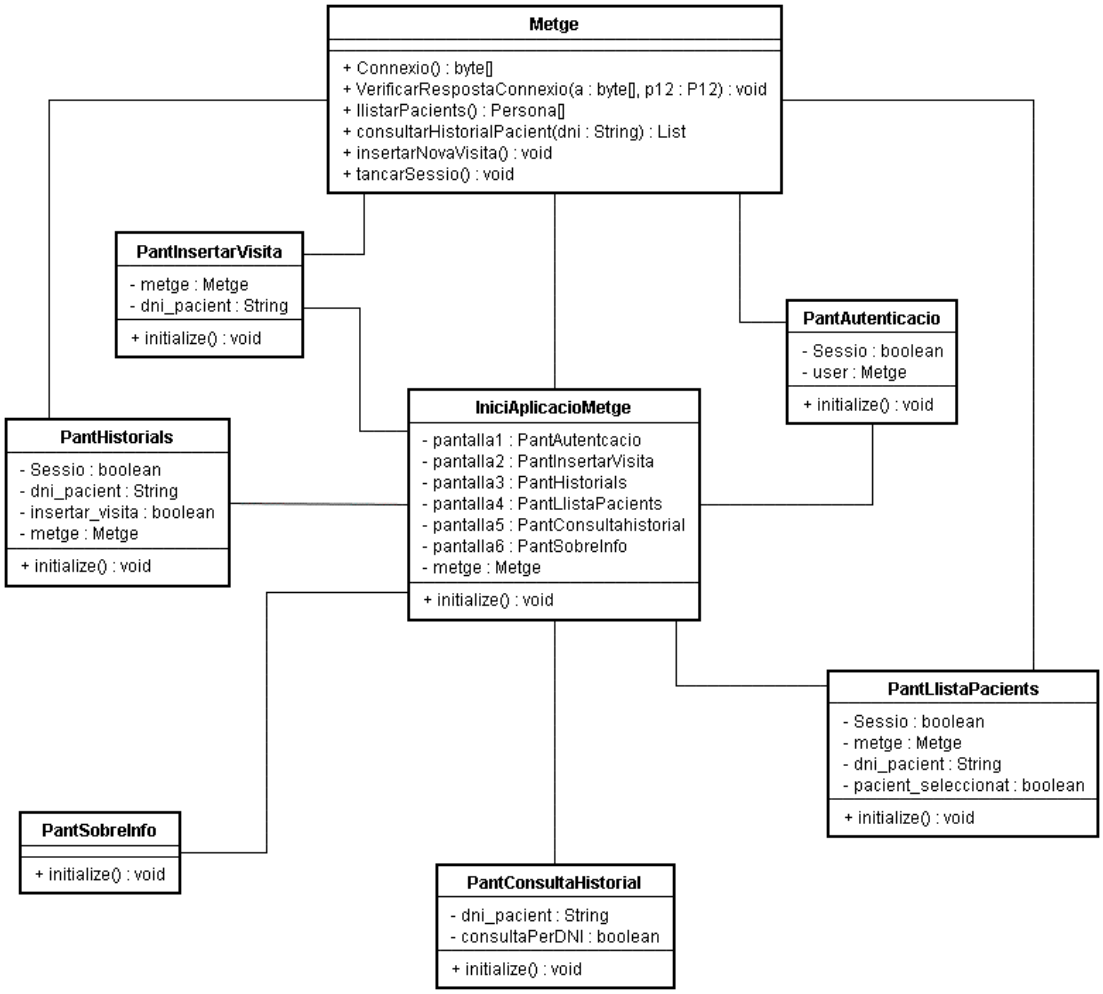

Figura 9-12. Diagrama de classes Interfície Gràfica – Metge

Del diagrama destaca la vista IniciAplicacioMetge, la qual conté la implementació de la pantalla principal del programa, i una representació de la resta de classes pantalla. Aquesta, serà qui controli, amb la inicialització i l'establiment de les propietats de visibilitat, la resta de classes en funció dels esdeveniments generats per l'usuari metge de l'aplicació.

Tanmateix, aquestes classes s'encarregaran de cridar a les funcions i mètodes de la classe Metge, i de preparar i de visualitzar el contingut dels resultats que s'han obtingut a partir de l'execució dels serveis.

9.5.2. Aplicació Pacient

El següent diagrama de classes reflecteix el conjunt de noves classes implementades i, de les seves relacions, per a formar l'aplicació pacient.

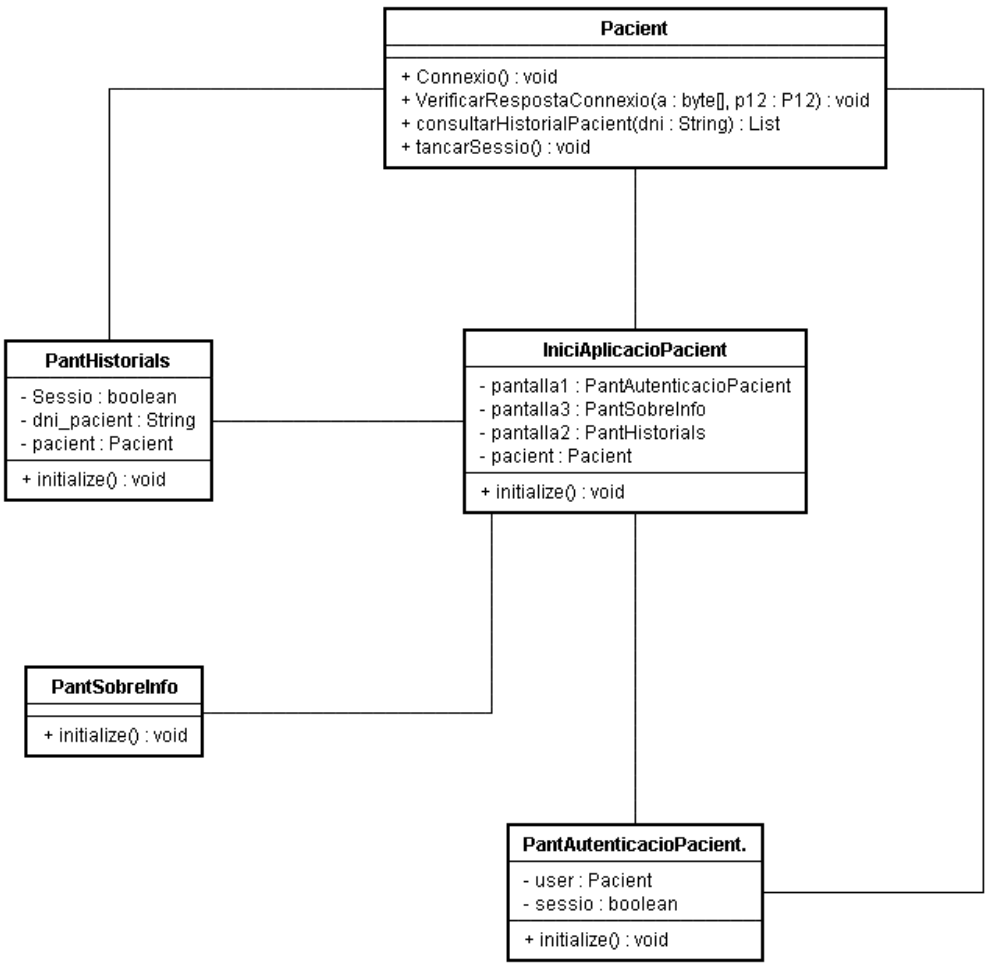

Figura 9-13. Diagrama de classes Interfície Gràfica - Pacient

De la mateixa manera que en el cas de l'aplicatiu Metge, també existeix una classe que podem anomenar central: IniciAplicacioPacient. Aquesta classe conté la vista de la pantalla principal i representacions de les demés vistes. Serà l'encarregada de controlar la visualització de les respectives vistes que formen el programa.

Tanmateix, la resta de classes s'encarregaran de cridar a les funcions i mètodes de la classe Metge, i de preparar i de visualitzar el contingut dels resultats que s'han obtingut a partir de l'execució dels serveis.

10. Joc de proves

10.1. Joc de proves I

Es realitzarà una prova d'inserció (per part del metge amb NIF 33333333-A) de dues visites per a un pacient. Cal inicialitzar l'aplicació referent als metges i seguir l'ordre de la seqüència descrita a continuació.

- Havent inicialitzat l'aplicació, Opcions → Autenticar-se
- NIF: 33333333-A, uoc0506, .../pki/Metge.p12
- Prémer Entrar
- Opcions → Llistar Pacients
- Seleccionar pacient amb NIF 22222222-A i prémer Veure Historial
- Prémer Inserir Visita
- Escriure dades diagnòstic, tractament i evolució
- Prémer Insertar
- Opcions → Llistar Pacients
- Seleccionar pacient amb NIF 22222222-A i prémer Veure Historial
- Prémer Inserir Visita
- Escriure dades diagnòstic, tractament, evolució i marcar Privat
- Prémer Insertar
- Opcions → Llistar Pacients
- Seleccionar pacient amb NIF 22222222-A i prémer Veure Historial
- Es veuran les dues visites insertades
- Prémer Tornar
- Tancar finestra o Opcions→ Finalitzar Sessió

10.2. Joc de proves II

A continuació, tornarem a inicialitzar l'aplicatiu metges. Aquesta prova consistirà en comprovar que es pot consultar l'historial d'un pacient no assignat al metge que s'autentica. Els passos a seguir són els següents:

- Havent inicialitzat l'aplicació, Opcions → Autenticar-se
- NIF: 23232323-A, uoc0506, .../pki/Metge2.p12
- Prémer Entrar
- Opcions → Consulta d'historials
- Introduir NIF 22222222-A i prémer Consultar
- El pacient no és un dels que té assignats el metge. No es pot Insertar cap Visita, ni tampoc apareixerà la visita privada.
- Prémer Tornar
- Tancar Finestra

10.3. Joc de proves III

A continuació, tornarem a inicialitzar l'aplicatiu metges. Aquesta prova consistirà en:

- Havent inicialitzat l'aplicació, Opcions → Autenticar-se
- NIF: 22222222-A, uoc0506, .../pki/Pacient.p12
- Prémer Entrar
- Opcions → Consulta de l'historial
- El pacient podrà consultar les dades corresponents al seu historial.
- Prémer Tornar
- Tancar Finestra

10.4. Proves d'error

- En cas de l'existència errònia en algun dels paràmetres introduïts en la pantalla d'autenticació, es mostrarà el següent missatge:

Figura 10-1. Error, usuari no autenticat

- En cas que s'hagi introduït un NIF de consulta erroni en la pantalla de Consulta per DNI, es mostrarà el següent missatge d'error.

Figura 10-2. Error, DNI del Pacient incorrecte

11. Conclusions

11.1. Conclusions generals

Una vegada finalitzat el procés d'elaboració del projecte, es pot afirmar que s'han assolit els objectius inicials proposats. S'ha aconseguit implementar un sistema capaç de gestionar historials mèdics de pacients a través d'una xarxa de comunicacions de forma segura.

S'han complert tots els objectius en quant a seguretat passant per vàries etapes en les que, primer, s'analitzaven i s'estudiaven els requeriments i les necessitats de protecció del sistema identificades i, després, s'elaborava el disseny i el desenvolupament dels esquemes criptogràfics. Així, s'han confeccionat mecanismes i funcions capaces de mantenir, preservar i protegir, l'accés a la informació davant qualsevol possible amenaça, complint les propietats d'autenticació, confidencialitat, integritat i no-repudi.

S'ha assolit l'establiment dels missatges que s'intercanvien entre els usuaris i el gestor i que aquests segueixin un format d'informació estructurada basada en *XML*. Tanmateix, s'ha aconseguit establir la comunicació remota entre els diferents components mitjançant la implementació de diferents classes que permeten obtenir les invocacions remotes.

S'ha dut a terme la gestió de la informació mitjançant la implantació d'una base de dades i el desenvolupament de classes encarregades d'executar mètodes per al seu tractament.

S'han dissenyat i implementat les interfícies gràfiques d'usuari amb les que els clients del sistema interaccionaran per a poder executar les funcionalitats demanades.

Finalment, es pot afirmar que s'ha obtingut una aplicació funcional en la que els seus usuaris, metges i pacients, poden dur a terme accessos al sistema i la consulta de determinada informació. Per la seva banda, els metges, també poden executar operacions configurades per a la modificació de certes dades. Tots aquests mètodes d'accés, consulta i modificació d'informació, són controlats per rigorosos mecanismes de seguretat i pel gestor central el qual esdevé peça clau en el gestió de la informació.

11.2. Conclusions personals

Com a experiència personal, puc dir, que aquest projecte ha suposat un gran repte a aconseguir, traduït amb un gran esforç i una dedicació intensa.

Per a desenvolupar aquest projecte ha estat necessari adquirir, aprofundir i descobrir una visió pràctica envers a la utilització d'algunes tecnologies i eines emprades, les quals han aportat nous coneixements. En particular, remarcar que he tingut l'oportunitat d'aprendre a implementar una aplicació utilitzant eines i tecnologies criptogràfiques amb les quals prèviament no havia tingut cap mena de contacte.

Tanmateix, he descobert nous aspectes sobre la programació amb *Java* com, per exemple, el dissenyar i confeccionar interfícies gràfiques d'usuari utilitzant la llibreria Swing. De la mateixa manera, he pogut profunditzar els coneixements que tenia sobre l'entorn de desenvolupament *Eclipse* i el seu editor visual.

Per tot això i, després de fer una reflexió pensant amb tot el treball que hi ha darrere d'aquest projecte i que, finalment, s'ha aconseguit el compliment dels objectius esmentats, puc dir, que el resultat generat es tradueix en una gran satisfacció personal.

12. Glossari

Tot seguit es presenta un recull i breu definició de diferents termes i paraules clau que han estat utilitzades en l'elaboració d'aquesta documentació.

API: *Application Programming Interface*, és el conjunt de funcions i procediments (o mètodes) que ofereix certa llibreria per a ser utilitzada per un altre software com una capa d'abstracció.

Autenticació: és la propietat que fa referència a la identificació. És el nexe d'unió entre la informació i l'emissor d'aquesta.

Autoritat de certificació: corresponent a les sigles CA, Certification Authority en anglès, és la responsable d'emetre i revocar els certificats. És l'entitat de confiança que dóna legitimitat a la relació d'una clau pública amb la identitat d'un usuari o servei.

Autoritat de registre: corresponent a les sigles RA, Registry Authority, és l'encarregada de verificar el lligam entre les claus públiques i la identitat dels seus titulars.

Base 64: és un sistema de numeració posicional que usa 64 com a base. És la major potència de dos que pot ser representada utilitzant únicament els caràcters imprimibles d'ASCII.

Criptografia: ciència i estudi de l'escriptura secreta.

Confidencialitat: propietat que assegura que només aquells que estan autoritzats tindran accés a la informació. Sovint a aquesta propietat també se la coneix amb el nom de privadesa.

Certificat digital: és una estructura de dades que conté informació del propietari de les claus criptogràfiques, la clau pública en si, i, una signatura digital dels dos camps anteriors que hi dóna validesa.

Desxifratge: procés de transformació del text xifrat en text en clar.

Eclipse: es tracta d'un entorn integrat de desenvolupament (*IDE*) obert i extensible.

IAIK: són les sigles corresponents a *Institute for Applied Information Processing and Communication*, desenvolupador d'una llibreria criptogràfica amb el mateix nom.

Integritat; és la propietat que assegura la no-alteració de la informació. Aquesta alteració pot ser inserció, esborrament o substitució de la informació.

Java: llenguatge de programació multi-plataforma, robust, interpretat, distribuït, portable, i orientat a objectes, desenvolupat per Sun Microsystems a principis dels anys 1990.

JDBC: o *Java Database Connectivity*, és una API que permet l'execució d'operacions sobre base de dades des del llenguatge de programació Java.

JDOM: API pensada específicament per al processament i la manipulació de documents XML amb Java.

MySQL: sistema de gestió de base de dades relacional, multi-fil i multiusuari, que utilitza el llenguatge SQL.

OpenSSL: és un projecte de software desenvolupat pels membres de la comunitat *Open Source* (per a lliure descàrrega), consistent en un robust paquet d'eines d'administració i llibreries relacionades amb la criptografia.

PKI: la infraestructura de clau pública (*Public Key Infraestructure*) és la combinació de maquinari, programari, persones, polítiques i procediments de seguretat que permeten l'execució amb garanties de les operacions criptogràfiques.

PKCS: les normes *PKCS* o public-key cryptography standards, són un conjunt d'especificacions desenvolupades en els laboratoris RSA amb l'objectiu d'establir una norma comuna a la indústria sobre els formats de les dades utilitzades en la criptografia de clau pública.

RMI: *Java Remote Method Invocation* és un mecanisme proporcionat en *Java* per a invocar mètodes de manera remota.

Subscriptors i entitats finals: els subscriptors i entitats finals són aquells que posseeixen un parell de claus (pública i privada) i un certificat associat a la clau pública.

SQL: (*Structured Query Language*) és un llenguatge estàndar de comunicació amb bases de dades relacionals.

Swing: biblioteca gràfica per a *Java* que proporciona accés de forma senzilla a tot un ventall ampli de components gràfics de programació.

UML: (*Unified Modeling Language*, Llenguatge de Modelat Unificat) és un llenguatge per especificar, dissenyar, construir i documentar sistemes, inicialment de programari orientat a objectes.

Xifra: mètode secret d'escriptura, mitjançant el qual un text en clar es transforma en un text xifrat.

Xifratge: procés de transformació d'un text en clar en un text xifrat.

XML: el *eXtensible Markup Language*, llenguatge de marques extensible, és un metallenguatge extensible, d'etiquetes, desenvolupat pel W3C.

13. Bibliografia

Detall de les fonts d'informació en format paper i digital a les que s'han accedit i consultat per a la confecció d'aquest projecte.

RMI

- Programació / Tutorials

  http://www.programacion.com/java/tutorial/rmi/

  http://www.chuidiang.com/java/rmi/rmi.php

- Tutorial oficial Java

  http://java.sun.com/j2se/1.5.0/docs/guide/rmi/

- Descripció general, Model d'Objectes Distribuïts

  http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/RMI1.html

JDOM API Specification

- Especificació de l'API JDOM

  http://www.jdom.org/docs/apidocs/index.html

JDBC

- Getting Started with the JDBC API

  http://java.sun.com/j2se/1.5.0/docs/guide/jdbc/getstart/GettingStartedTOC.fm.html

Swing

- Swing features – tutorials

  http://java.sun.com/docs/books/tutorial/ui/features/index.html

  http://java.sun.com/docs/books/tutorial/uiswing/index.html

Visual Editor

- Instal·lació de Visual Editor d'Eclipse

  http://www.ajpdsoft.com/modules.php?name=News&file=article&sid=271

XML

- XML Information Set (Febrer 2004)

  http://www.w3.org/TR/2004/REC-xml-infoset-20040204/

OpenSSL

- Pàgina Web oficial

  http://www.openssl.org/

PKCS#12

- PKCS #12: Personal Information Exchange Syntax Standard

  http://www.rsa.com/rsalabs/node.asp?id=2138

PKI

- Inraestructura de clau pública

  http://www.eurologic.es/soluciones/que-es-pki.htm

MySQL

- MySQL 5.0 Reference Manual

  http://dev.mysql.com/doc/refman/5.0/es/index.html

- Eines

  http://dev.mysql.com/downloads/

Eclipse

- Eclipse – an open development platform

  http://www.eclipse.org/

14. Annexos

14.1. Relació dels arxius adjunts a la memòria

A continuació es detalla el contingut de l'entrega del projecte.

Carpeta	Contingut	Descripció
/src	- classes .java - script creació BD - script creació permisos BD	Codi font del sistema més arxius per a la creació de la base de dades i els corresponents permisos.
/project	Projecte Eclipse	Exportació del projecte desenvolupat amb Eclipse.
/pki	- arxius P12 creats - arxius crt (certificats) - llegeix-me.txt	Tots els arxius PKCS#12 i els certificats dels usuaris utilitzats per a les proves. A més inclou el fitxer de configuració utilitzats per a generar-los.
/doc	- PFC_Memoria.doc - PFC_Memoria.pdf	Documentació del projecte.
/bin	- arxius .class - arxius .bat - llegeix-me.txt	Tots els binaris necessaris per a l'execució. Arxius .bat que faciliten l'execució de l'aplicació. Arxiu de text indicant quins canvis caldrien realitzar per a l'execució.

Figura 13-1. Relació d'arxius PFC

14.2. Instal·lació del sistema

14.2.1. Instal·lació *IAIK PKI*

Els passos per a instal·lar *IAIK* són aproximadament els que segueixen a continuació:

- descarregar la darrera versió del *JDK* de *SUN* i instal·lar-lo: http://java.sun.com/javase/downloads/index.jsp

- descarregar la darrera versió de IAIK. Cal registrar-se però no suposa cap cost. Descarregar l'arxiu iaik_jec_full.jar del següent enllaç: http://jce.iaik.tugraz.at/download/evaluation/index.html

- descarregar les polítiques de seguretat de java que permeten emprar qualsevol longitud de clau (Java Cryptography Extension – JCE) en relació a la versió del JDK instal·lada.

- (Windows) copiar l'arxiu iaik_jce_full.jar als directoris:

  - C:\Archivos de Programa\Java\jdk1.6.0.2\jre\lib\ext
  - C:\Archivos de Programa\Java\jdk1.6.0.2\lib\ext

- (Linux) copiar l'arxiu iaik_jce_full.jar al directori:

  - $JAVA_HOME/jre/lib/ext

- (Windows) dins de l'arxiu jce_policy-6.zip hi ha els arxius:

  - local_policy.jar
  - US_export_policy.jar

  Copiar-los a:

  - C:\Archivos de Programa\Java\jdk1.6.0.2\jre\lib\security
  - C:\Archivos de Programa\Java\jdk1.6.0.2\lib\security

- (Linux) dins de l'arxiu jce_policy-6.zip hi ha els arxius:

  - local_policy.jar
  - US_export_policy.jar

  Copiar-los a:

  - $JAVA_HOME/jre/lib/security

14.2.2.Instal·lació *OpenSSL*

Els passos per a instal·lar *OpenSSL*, són els següents:

- (Windows) descarregar l'executable i instal·lar: http://www.openssl.org/source/

- (Linux) descarregar el codi font i compilar: http://www.openssl.org/related/binaries.html

14.3.3. Instal·lació *JDOM*

Els passos per a instal·lar *JDOM*, són els següents:

- Descarregar la versió de *JDOM* de http://www.jdom.org/dist/binary/
- Descomprimir el fitxer
- Executar l'arxiu build.bat
- Situar jdom.jar de manera que *Java* el detecti en el *classpath*

14.3.4. Instal·lació de MySQL

- Descarregar l'arxiu d'instal·lació de la Web: http://dev.mysql.com/downloads/mysql/5.0.html#downloads
- Seguir els passos d'instal·lació de la Web de referència: http://dev.mysql.com/doc/refman/5.0/es/installing.html

14.3.5. Instal·lació de JDBC de MySQL

- Descarregar l'arxiu mysql-connector-java-5.1.5.zip de: http://dev.mysql.com/downloads/connector/j/5.1.html
- Un cop descarregat, s'ha de situar dins del *classpath*

14.3.6. Creació de les taules buides de la base de dades

- De l'entrega del projecte, anar al directori \bin i amb el navegador *MySQL* obrir l'arxiu Script_creacio_BD.sql i executar-lo. Es crearà una base de dades buida i un usuari de la base de dades "administrador" amb contrasenya "password".

14.3.7. Inserció de les dades de prova

Executar la classe JDBCDades existent al directori bin\src_servidor\aplicacio:

- bin\src_servidor\> java aplicacio.JDBCDades

O bé executar l'arxiu InsertarDades.bat inclòs en el directori \bin del projecte.