TFC: Missatgeria Instantània

Amb el suport de LaCOLLA

Josep Tomàs Comellas
ETIS

Xavier Vilajosana Guillen

10 de Gener de 2005

Índex

0. Resum descriptiu

Avui en dia estem acostumats a treballa de forma quotidiana amb programes de missatgeria instantània. Generalment el Messenger, tot hi que en l'actualitat n'han aparegut de nous amb similars característiques.

Aquest tipus de programes fan possible la comunicació entre usuaris, els quals poden estar situats en qualsevol lloc del món, gràcies la Internet i la infrastructura que aquesta proporciona.

El projecte presentat pretén endinsar-se en aquest concepte i proporcionar un valor afegit: l'eliminació dels punts intermitjos de fallida. Aquesta millora sobre d'altres solucions recau en la capacitat de treballar sense la necessitat d'un punt central d'unió entre els nodes que participen en la comunicació.

Per fer possible aquest objectiu, s'ha dissenyat dins d'aquest projecte l'aplicatiu anomenat "Tuté", el qual ha estat desenvolupat en Java i que interactua amb el motor de LaCOLLA, aquesta es pot descriure com una infrastructura de comunicació descentralitzada.

Aquesta aplicació ha estat dissenyada a partir de quatre línies d'estudi:

1.- Desenvolupament en llenguatge Java:
Tota l'aplicació ha estat codificada en el llenguatge de programació de "Sun Microsystems" que ofereix la característica de poder ser utilitzada en qualsevol plataforma.

2.- Descobriment, connexió i comunicació amb LaCOLLA:
La comunicació entre l'aplicació i LaCOLLA es duu a terme a través de l'Api del motor.

3.- Descobriment, disseny i adaptació de l'entorn gràfic:
El llenguatge de desenvolupament Java proporciona diferents vies de programació d'entorns gràfics, entre els quals s'ha triat l'estendard swing.

4.- Disseny i desenvolupament de la comunicació LaCOLLA ⇔ Aplicació:
A fi de treballar de forma estandaritzada a la hora de intercanviar missatges, s'ha optat per implementar la solució sobre un suport XML.

1. Cos de la memòria

En els següents apartats es detalla el disseny i els procediments requerits per tal de duu a terme aquest projecte.

1.1. Introducció

Tot seguit es descriu la raó de ser del projecte presentat.

1.1.1. Justificant del TFC i context en el qual es desenvolupa

L'elecció d'aquest projecte com a punt final de la carrera d'Informàtica de Sistemes pren sentit arrel del meu propi interès sobre el món de la informàtica distribuïda.

Hi ha hagut dues grans raons per la tria realitzada: en primer lloc, el suport en el que es basa el projecte: la programació en Java i tot el que això representa. I en segon lloc, el desconeixement i per tant l'inquietut de treballar en solucions que puguin funcionar a través de la xarxa Internet.

1.1.2. Objectius del TFC

L'objectiu primari del projecte recau en la capacitat de connectar dos nodes amb connexió a xarxa, en tant sigui una simple LAN com a través d'Internet, i fer possible establir-hi una comunicació bidireccional utilitzant el concepte de missatgeria instantània.

L'objectiu secundari recau en l'estudi del motor de comunicació distribuïda que brinda LaCOLLA, així com la connexió de l'aplicatiu desenvolupat amb aquesta solució a fi d'emprar-lo com a suport de comunicació.

1.1.3. Enfocament i mètode seguit

Com ja es descriu en el resum inicial, s'ha enfocat el projecte a partir de 4 línies de treball que es van col·lapsant fins aconseguir l'aplicatiu final.

1. – *Llenguatge Java*: Ampliació dels coneixements obtinguts en la carrera en el desenvolupament i el disseny de l'estructura de classes que permeti un bon aprofitament del avantatges que ofereix la programació orientada a objectes.

2.– *LaCOLLA*: Estudi i implementació d'aquesta solució a fi d'utilitzar-la com a plataforma de comunicació capaç de permetre separar la capa on es situa l'aplicatiu desenvolupat de la capa de xarxa.

3.– *Entorn gràfic*: Descobriment de l'entorn de disseny i desenvolupament NetBeans i de les eines que proporciona Java per tal d'obtenir solucions en entorn gràfic.

4.– *Comunicació LaCOLLA ⇔ Aplicació*: Estudi, disseny i desenvolupament de les classes que permeten fer possible la comunicació entre l'aplicatiu desenvolupat i l'Api de LaCOLLA.

La darrera via de treball ha implicat l'estudi i l'implementació de les tecnologies RMI i XML.

1.1.4. Planificació del projecte

En un principi es va establir una temporalitat a fi de delimitar cada línia de treball. Però en la realitat es va haver d'anar canviant entre els diferents fronts finalitzant el projecte amb un treball paral·lel de quasi tots els fils.

Globalment es va seguir la següent cronologia:

1.- Estudi de la programació en Java orientada a la interfície gràfica i el disseny oportú de el cas concret de l'aplicació.
Arrel d'aquest punt es va canviar l'eina de desenvolupament utilitzada, "l'eclipse", per un altre de més potencia i comoditat: el Netbeans.

2.- Recepció de l'api de LaCOLLA i instal·lació del producte.
Descobriment del funcionament i primera fase de la implementació de la classe de vincle entre l'aplicació i el motor de LaCOLLA.

3.- Es decideix d'emprar XML en l'intercanvi de missatges.
Disseny del bloc XML: primera fase d'implementació i primeres proves.

4.- Disseny de l'estructura de classes. Primera fase d'implementació del codi que comunica l'aplicació i LaCOLLA.

5.- Implementació del bloc XML en la comunicació entre l'aplicació ⇔ LaCOLLA ⇔ l'aplicació.
Redisseny del bloc XML per tal d'obtenir objectes serialitzables.

6.- Implementació definitiva de l'aplicació.
Modificació definitiva de l'entorn gràfic.
Afegida funcionalitat XML per l'intercanvi d'imatges.

7.- Implementació de les funcionalitats d'alta d'usuaris i grups.
Modificació de l'entorn gràfic implicat en la nova funcionalitat.

1.1.5. Productes obtinguts

En finalitzar el disseny i la implementació del projecte, s'ha obtingut una aplicació d'intercanvi de missatges escrits així com petites imatges entre usuaris connectats en xarxa, tant sigui local com global.

Aquesta aplicació pot considerar-se com una implementació en codi obert del l'entorn distribuït i descentralitzat que ofereix LaCOLLA.

1.1.6. Breu descripció dels altres capítols de la memòria

En els següents capítols es descriuen les classes utilitzades així com la seva relació.

També es descriu la funcionalitat de l'aplicació des de els diferents punts de vista.

1.2. Plataforma

En els següents punts de la memòria es descriuen les funcionalitats implícites de l'aplicació creada.

1.2.1. Independència del Sistema Operatiu

La implementació del codi de l'aplicació en llenguatge Java versió 1.4.1, la qual cosa implica que aquesta podrà ser transportable a qualsevol plataforma que implementi un Run-Time de Java.

La portabilitat descrita en el paràgraf anterior està subjecte a l'existència d'un "package" anomenat xerces.jar. La qual permet implementar solucions basades en XML.

1.2.2. Requeriments de l'aplicació

L'aplicació creada es basa en el motor de LaCOLLA, per la qual cosa es necessari que aquest estigui en funcionament i de coneixen l'adreça i els ports amb els es publica. En concret cal conèixer els ports corresponents a la instància i a l'api.

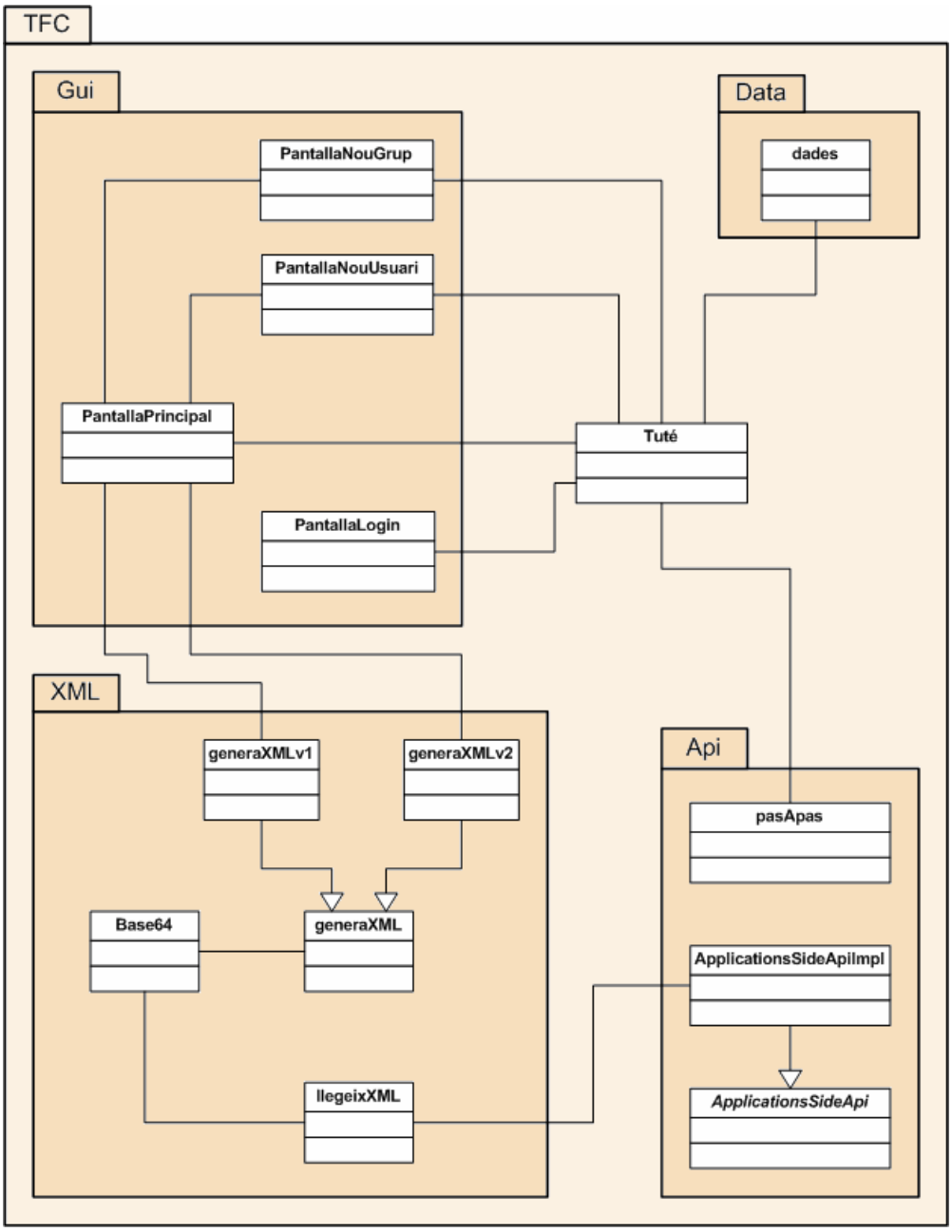
L'estructura de fitxers de l'aplicació cal que estigui situada dins del directori Apps de LaCOLLA.

1.2.3. Estructura de fitxers i breu descripció

Carpetes			Fitxers	Descripció
Apps				
↳	TFC		Tute	classe inicial
	↳ Api		ApplicationsSideApiImpl	classe publicada per el RMI i vinculada a LaCOLLA
			pasApas	classe d'implementació de les funcions de LaCOLLA
	↳ Data		dades	classe d'enmagatzemament de dades
	↳ Gui		PantallaLogin	interfície gràfica per el registre de l'usuari
			PantallaPrincipal	interfície gràfica principal de l'aplicació, i enviament de missatges i imatges
			PantallaNouGrup	interfície gràfica de petició de nous grup
			PantallaNouUsuari	interfície gràfica de petició de nous usuaris
		↳ Emoticons	... imatges varies ...	imatges (emoticons) enviables conjuntament amb els missatges
		↳ Imatges	... imatges varies ...	imatges de l'entorn gràfic
	↳ XML		Base64	classe recopilada per Internet de transformació en base64
			generaXML	classe pare per la generació d'objectes XML
			generaXMLv1	classe filla per la generació d'objectes XML destinada a traspàs de text
			generaXMLv2	classe filla per la generació d'objectes XML destinada a traspàs de text més imatges
			llegeixXML	classe de recuperació del contingut del objectes XML

1.2.4. Diagrama de classes

L'esquema següent mostra les relacions entre les diferents classes que intervenen en l'aplicació.



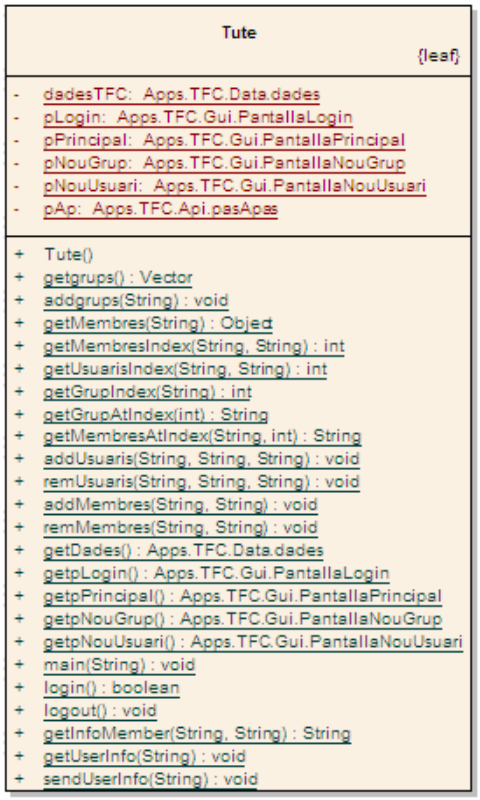
1.2.5. Estructura de classes

L'aplicació s'ha dissenyada seguint una estructura de directoris en referència a la funcionalitat que desenvolupen.

El nivell principal correspon a la carpeta Apps proporcionada per la pròpia instal·lació de LaCOLLA con a repositori d'aplicacions.

1.2.5.1. Nivell Apps.TFC

En aquest nivell hi ha situada la classe principal anomenada *Tute.java*, encarregada de la gestió de les peticions internes de l'aplicació.



Aquesta classe és la que centralitza les peticions des de i cap l'entorn gràfic així com des de i cap a l'api de LaCOLLA.

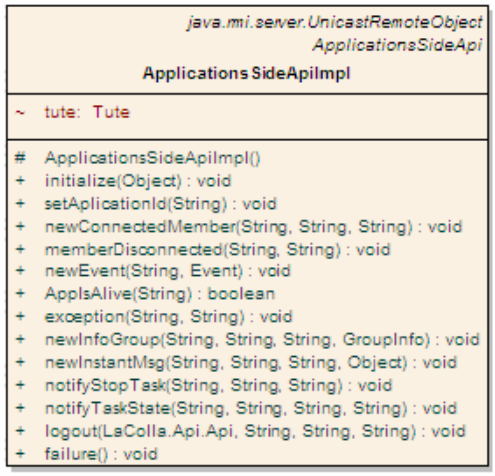
També es centralitza l'accés a la informació que reté l'aplicació.

Per el que fa als mètodes d'aquesta classe, hi estan definits totes aquelles funcions necessàries, com ara l'enllaç amb LaCOLLA i que prèviament cal formatar.

Bàsicament, aquesta classe actua de capa separadora entre les peticions que realitza l'interfície d'usuari i la capa més directament relacionada amb LaCOLLA i la seva api.

1.2.5.2. Nivell Apps.TFC.Api

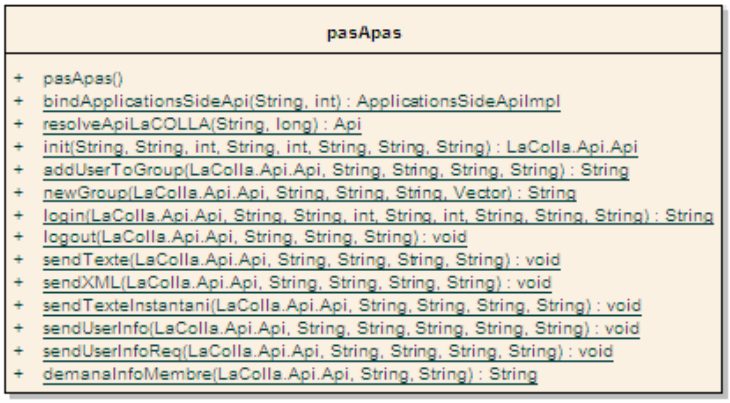
En aquest nivell hi ha implementades dues classes que efectuaran tasques de comunicació directe entre l'aplicació i l'api de LaCOLLA.



En primer lloc la classe *ApplicationSideApiImpl.java*, implementa la interfície *ApplicationSideApi*, i que desenvolupa la tasca de rebre totes les comunicacions de LaCOLLA gràcies a la utilització de la tecnologia RMI.

Un cop implementada aquesta classe cal cridar l'executable *RMI.exe* passant-li com a paràmetre la pròpia classe a fi de que es generin els fitxers *ApplicationSideApiImpl_Skel.class* i *ApplicationSideApiImpl_Stud.class*. Les qual possibiliten la instanciació remota d'aquesta classe.

Per altra banda hi ha la classe *pasApas.java*.



Aquesta classe implementa les funcions de comunicació entre LaCOLLA i l'aplicació, corresponent a les peticions d'informació i l'enviament dels propis missatges.

1.2.5.3. Nivell Apps.TFC.Data

En aquest nivell tan sols hi ha una classe implementada anomenada *dades.java*.

Aquesta classe es correspon al repositori de totes les dades que ha de mantenir en memòria l'aplicació i que tan sols es tindrà accés a través de l'instanciació i posterior publicació realitzada des de la classe Tute.java.

1.2.5.4. Nivell Apps.TFC.Gui

Aquest nivell està destinat a agrupar les implementacions corresponents a la interfície gràfica mostrades en l'apartat 1.2.4.

1.2.5.4.1. Nivell Apps.TFC.Gui.Emoticons

En aquest nivell no hi ha implementada cap classe. Tan sols hi ha ubicades les imatges (emoticons) que es poden enviar conjuntament amb els missatges de text.

1.2.5.4.2. Nivell Apps.TFC.Gui.Imatges

En aquest segon nivell també emmagatzema imatges, destinades aquest cop al propi aspecte de l'entorn gràfic.

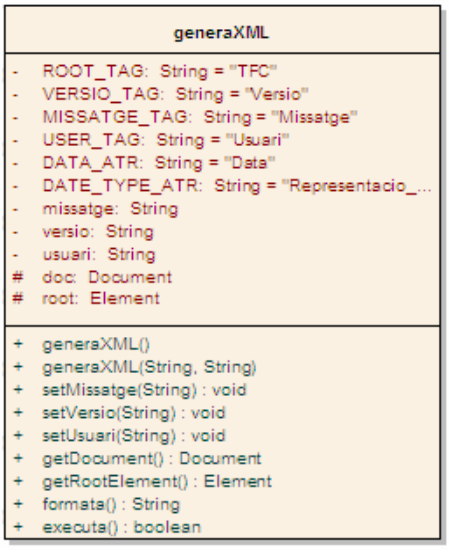
1.2.5.5. Nivell Apps.TFC.XML

Aquest nivell conté les classes que implementen la creació, i posterior restauració de les dades que es transmeten entre nodes en format XML.

Aquest bloc es pot dividir en dos apartats: el primer d'ells està destinat a la creació de l'objecte XML.

Donat per una banda el requeriment de l'aplicació d'utilitzar XML per el traspàs de dades i per altra banda la necessitat de treballar amb objectes serialitzables (requeriment del funcionament de l'entorn RMI), es va triar per implementar la solució lliure que publica SUN Microsistems anomenada *Xerces* i que s'inclou en el paquet Xerxes.jar (únic element no estàndard de l'aplicatiu).

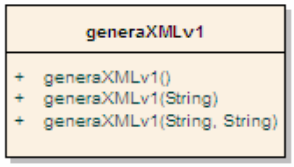
Per implementar aquest bloc s'ha creat una classe pare que implementa les funcionalitats bàsiques de la creació de l'XML anomenada *generaXML.java*:



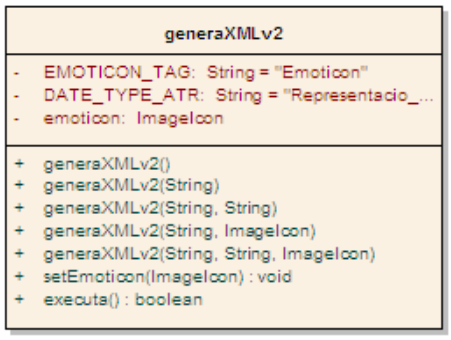
Aquesta classe s'encarrega del muntatge de l'objecte i la posterior serialització a fi de ser enviada per la xarxa.

Però donada la possibilitat de haver de poder "omplir" l'objecte amb diferents tipus de dades, s'han definit dues classes filles que hereten de l'anterior i que s'instancien segons les necessitats.

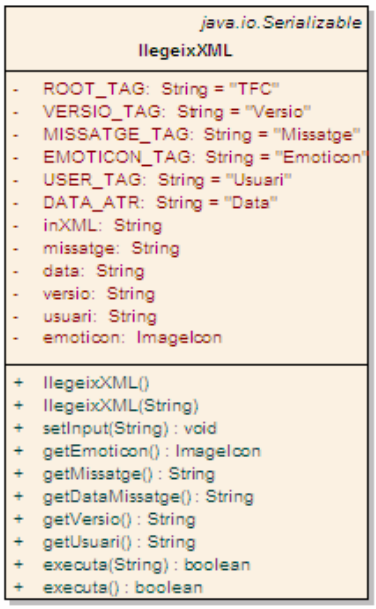
En primer lloc, es defineix la classe filla anomenada *generaXMLv1.java*. Aquesta serà utilitzada quan sigui necessari incloure en l'XML tan sols un missatge de text.



Per altra banda, la segona classe filla anomenada *generaXMLv2.java*, caldrà ser instanciada en el moment de voler enviar apart del propi missatge de text una petita imatge (emoticon).



El segon bloc d'aquest nivell correspon a la recuperació de l'objecte XML i la lectura dins d'aquest del missatge de text i en cas necessari de la imatge enviada.



La classe encarregada d'aquesta funció s'anomena *llegeixXML.java*.

Durant el procés de desenvolupament i posteriors proves, es van detectar errors durant l'enviament de caràcters especials. Aquest caràcters eren rebuts de forma incorrecte per el destinatari del missatge.

Per tal de solucionar aquest problema es va triar la opció de transformar els texts dels missatges en codificació base64.

La classe que es mostra sobre aquestes línies realitza la funció de codificar i descodificar les cadenes de bytes que calgui.

Posteriorment, també es va optar per emprar aquesta codificació en l'enviament d'imatges.

La classe *Base64.java* que s'ha utilitzat no ha estat desenvolupada en aquest projecte, sinó que s'ha utilitzat una implementació obtinguda d'Internet de la web *sourgeFORGE.net*.

1.3. Funcionament

En els apartats següents es descriu el funcionament de l'aplicació des de els diferents punts de vista: Usuari i Sistema.

1.3.1. Funcionament intern de LaCOLLA

El funcionament intern de LaCOLLA així com les seves necessitats a nivell de configuració estan descrites per el programador i adjuntades en l'apartat d'annexos.

1.3.2. Funcionament bàsic de l'entorn

Per tal de poder treballar amb aquesta aplicació, cal en primer lloc preparar l'entorn de treball.

1.3.2.1. Instal·lació de l'entorn Java

Tota l'aplicació ha estat desenvolupada en Java, al igual que el motor de LaCOLLA, la qual cosa implica que caldrà tenir instal·lada en l'estació el runtime de Java en una versió no inferior a la 1.4.1.

Aquest programari pot ser descarregat de la web del fabricant mostrada en l'apartat de bibliografia.

1.3.2.2. Instal·lació de LaCOLLA

Aquest entorn de treball es basa bàsicament en l'activació del motor de LaCOLLA.

La instal·lació del motor es pot realitzar executant el fitxer *LaCOLLA-final-retail-win32.exe* facilitat per el consultor.

L'opció més adient serà instal·lat tots els components que ofereix.

Un d'aquests components és la base de dades lliure MySQL, la qual en ser instal·lada conjuntament amb el resta de components ja vindrà correctament configurada i amb tots els complements i connectors necessaris.

Un cop finalitzada la instal·lació cal registrar el servei del servidor SQL MySQL. Aquest procés es duu a terme llençant una consola de MsDos, situar-se en el directori *bin* dins de l'estructura de directoris del servidor SQL i executar la següent comanda: *mysql-max-nt.exe --install*. Aquest pas crearà un servei en l'estació windows que s'haurà d'iniciar amb la comanda *net start MySQL*.

1.3.2.3. Configuració i posta en marxa de LaCOLLA

En haver instal·lat LaCOLLA, tenim la plataforma preparada, però no configurada.

Per tal de realitzar la configuració caldrà seguir el manual que s'inclou en la distribució: "*[BaseInstal·lació]\LaCOLLA\doc\starting with LaCOLLA-català .pdf*."

Tal com indica el manual, cal executar el motor de LaCOLLA llençant els fitxers de los *uoc1.bat* i *uoc2.bat*(un cop configurats).

El primer cop que s'inicien, realitzen certes tasques de configuració que poden provocar algun retard en les respostes als possibles clients.

1.3.3. Funcionament bàsic de l'aplicatiu

Un cop iniciat l'entorn de treball, ja es pot iniciar l'aplicatiu desenvolupat en aquest projecte.

Per tal de realitzar una prova de forma correcte, caldrà haver creat dos usuaris a través del fitxer per lots *configure.bat*.

Per tal de començar a conèixer l'entorn del programa, en els següents apartats es mostra l'aspecte i la funcionalitat de l'entorn gràfic

1.3.3.1. Interfície gràfica

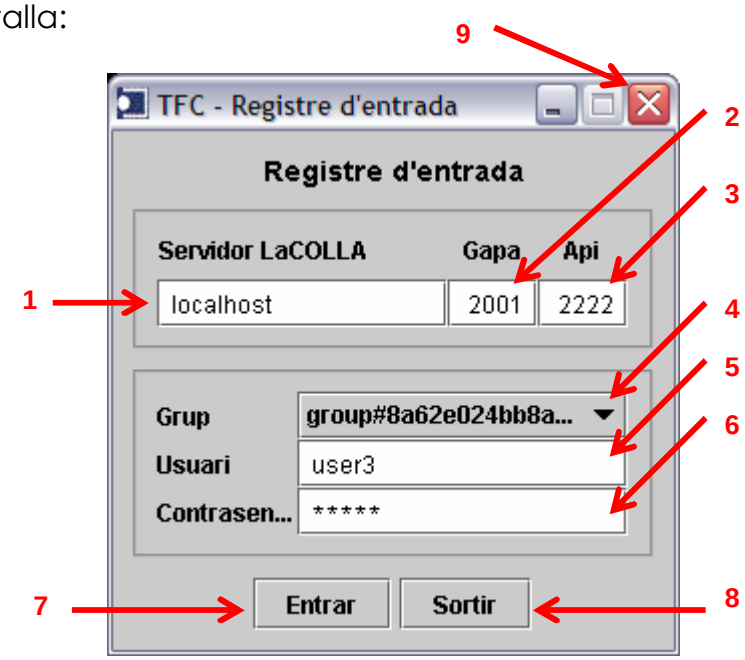
Totes les pantalles creades han estat implementades amb al solució de Java *swing*. Aquest paquet està integrat en la distribució estàndard 1.4.1, la qual cosa fa possible la portabilitat de l'aplicació.

Com ja s'ha indicat, totes les pantalles han estat dissenyades amb l'entorn de desenvolupament lliure NetBeans.

1.3.3.1.1. Pantalla Login

Nom de la classe: *Apps.TFC.Gui.PantallaLogin.java*

Pantalla:



Funcionament:

En iniciar l'aplicació, apareix la finestra anterior on cal especificar l'adreça on respon el servidor de LaCOLLA (1), el port on és publica el Gapa (2) finalment el port on es publica l'api (3).

Tot seguit es troba l'identificador que LaCOLLA assigna al grup al qual s'hi desitja autentificar (4), aquest paràmetres serà llegit per l'aplicació en un principi o bé rebut posteriorment durant el funcionament normal i emmagatzemat en el fitxer de configuració. Per últim, l'usuari (5) i la contrasenya corresponent (6).

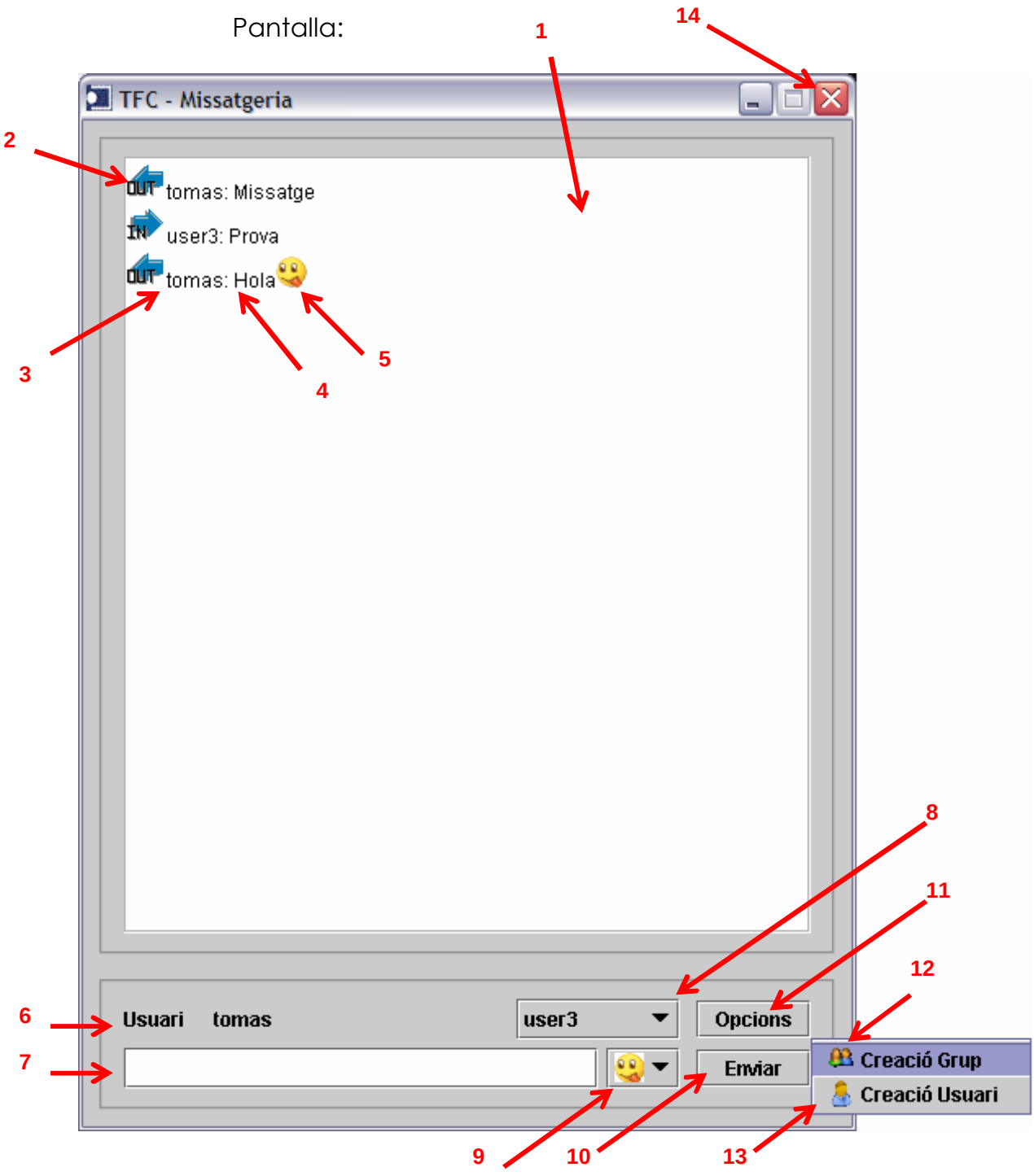
Un cop omplerts tots els camps, es pot prémer el botó *Entrar* (7), el qual iniciarà el procés de comunicació amb LaCOLLA i l'accés al resta de l'aplicació un cop acceptades les credencials.

En qualsevol cas, es podrà prémer el botó *Sortir* (8) o bé la creu de la barra superior (9) per finalitzar l'aplicació. En aquest cas no s'establirà cap comunicació amb LaCOLLA.

1.3.3.1.2. Pantalla Principal

Nom de la classe: *Apps.TFC.Gui.PantallaPrincipal.java*

Pantalla:



Funcionament:

Un cop acceptades les credencials, la pantalla de registre s'amagarà automàticament i es mostrarà la finestra anterior, la qual correspon a la interfície pròpia de treball que té la funcionalitat bàsica de permetre enviar i rebre missatges i petites imatges.

En el part superior de la finestra es troba el bloc del missatges (1), on es mostra per cada una de les comunicacions: un icona indicant gràficament el sentit del missatge (2), l'usuari que l'ha escrit (3), el propi missatge (4) i opcionalment, una petita icona (5).

Ja en el bloc inferior, s'hi agrupen la part d'interacció amb l'usuari.

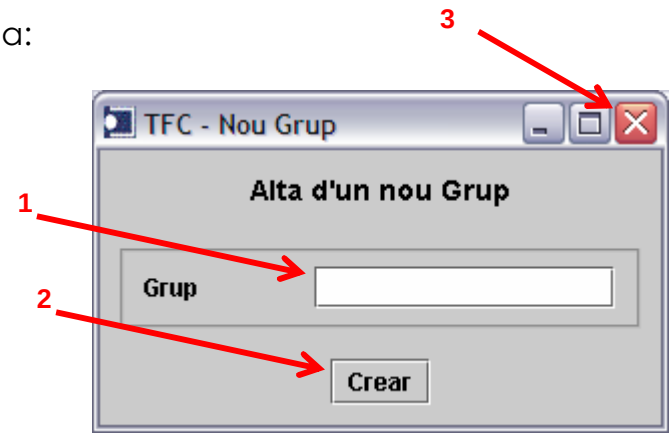
En aquest bloc es mostra en primer lloc l'usuari connectat (6), tot seguit hi ha el quadre de text on es pot escriure el missatge que es vol enviar (7). Tot seguit cal seleccionar de la llista l'usuari de destí (8), el qual s'ha afegit automàticament en detectar la entrada d'aquest. Opcionalment, es pot triar una imatge de la llista (9) i finalment, amb el botó d'enviar (10) s'inicia el procés d'enviament del missatge el qual es mostrarà en el bloc superior tant per usuari que l'envia com per el que el rep.

Per últim, en prémer el botó Opcions (11), es desplega un petit menú que permet crear grups (12) o bé usuaris (13).

1.3.3.1.3. Pantalla creació Grups

Nom de la classe: *Apps.TFC.Gui.PantallaNouGrup.java*

Pantalla:



Funcionament:

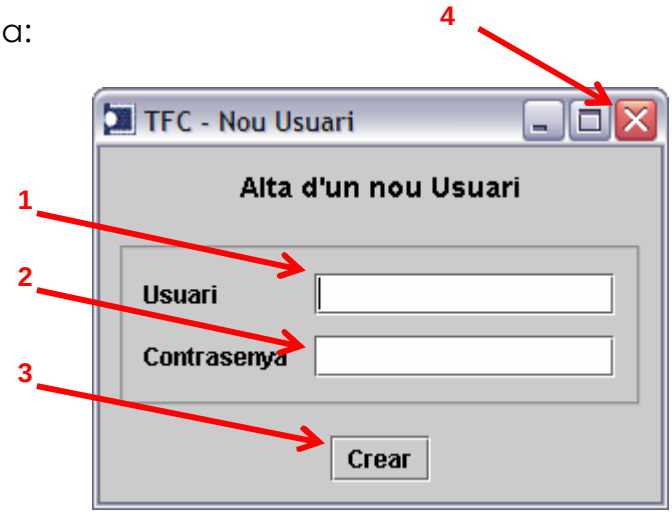
Per tal d'afegir nous grups a la base de dades de LaCOLLA cal emprar aquesta pantalla en la qual es pot introduir el nom desitjat en el camp Grup (1), i posteriorment prémer el botó Crear (2) a fi de registrar la petició en el sistema.

En el cas de voler cancel·lar la opció de creació caldrà tancar la finestra amb la creu de la part superior (3).

1.3.3.1.4. Pantalla creació Usuaris

Nom de la classe: *Apps.TFC.Gui.PantallaNouUsuari.java*

Pantalla:



Funcionament:

Per tal d'afegir nous usuaris en la base de dades de LaCOLLA cal emprar aquesta pantalla en la qual es pot introduir el nom de l'usuari en el primer camp de text (1), la contrasenya corresponent en el camp inferior (2) i finalment prémer el botó Crear (3) a fi de registrar la petició en el sistema.

En el cas de voler cancel·lar la opció de creació caldrà tancar la finestra amb la creu de la part superior (4).

1.3.4. Funcionament intern de l'aplicatiu

L'aplicació desenvolupada en aquest projecte es pot descriure bàsicament com una implementació en codi obert que utilitza la plataforma distribuïda i descentralitzada que ofereix LaCOLLA.

Aquesta descripció implica que la major part de la dificultat que puguin haver aparegut es deuran a la integració i comunicació entre tots dos entorns.

Arrel d'això aquest punt es centrarà en l'explicació del funcionament de l'aplicació en aquells aspectes que afectin directament al concepte plantejat.

1.3.5. Problemàtica 1: registre en LaCOLLA

LaCOLLA, com a sistema integrat de tractament d'usuaris i grups, requereix d'una autentificació en el seu sistema. La qual cal realitzar-la via la funció *login()* de *l'Api*, i sempre tenint en compte de guardar-ne el valor retornat, dons identifica la sessió establerta entre l'aplicació i LaCOLLA.

Per tal de fer aquesta aplicació transportable d'un equip a un altre, s'ha triat per llegir certa informació dels propis fitxers de configuració del motor de LaCOLLA. Aquest fet implica dos aspectes a tenir en compte: el programa ha d'estar sempre ubicat directament sota la carpeta *Apps* creada en la instal·lació. L'altre aspecte correspon a la facilitat afegida a l'entorn, donat que no caldrà buscar quin és l'identificador del grup per defecte ni l'identificador del Gapa on cal realitzar la connexió.

1.3.6. Problemàtica 2: informació de registre

A partir del primer registre correcte en l'aplicació, es crea un fitxer de configuració anomenat *dadesTFC.ini* i situat en la carpeta principal de LaCOLLA.

Aquest fitxer contindrà les dades utilitzades en l'últim registre correcte, i la llista de tots els grups creats.

1.3.7. Problemàtica 3: personalització de la informació

Per tal de poder utilitzar uns valors personalitzats a l'hora d'iniciar una nova sessió, es poder passar aquestes dades per paràmetres, indicar-les en el fitxer de configuració descrit en l'apartat anterior, o bé introduir-los directament en la finestra de login.

Els paràmetres són:

/U:[usuari]
/C:[Contrasenya]
/H:[Adreça de la instància de LaCOLLA]
/P:[Port del Gapa de LaCOLLA]
/A:[Port de l'Api de LaCOLLA]
/L:[Port de l'aplicació]
/G:[Identificador del Grup generat per LaCOLLA]

1.3.8. Problemàtica 4: obtenció de l'usuari associat al membre

Cada cop que un usuari es registra correctament en LaCOLLA, aquesta ho comunica a cadascun del altres usuaris. La forma de comunicar aquest fet és enviar un missatge indicant el membre associat al l'usuari registrat.

Aquest fet comporta dues problemàtiques: la primera ve donada en el fet que tots els missatges rebuts per LaCOLLA són notificats a l'usuari cada cop que aquest inicia la instància, però mai s'esborren. L'altre problema recau en la dada rebuda, la qual no és últim per l'usuari que interactua.

La solució ha passat per establir un petit protocol de comprovació de l'existència d'un client en ser notificat el fet per el sistema i així mateix la petició de l'usuari corresponent a la identificació de membre rebuda.

El procediment consisteix en enviar un missatge instantani al membre indicat, enviant-li com a text el caràcter "**?**". Aquest tipus de missatges no queden desats en la base de dades de LaCOLLA sinó que són enviats directament al membre indicat. Si aquest està actiu i rep el missatge el respondrà amb un altre missatge instantani introduint en el camp de text el caràcter "***!***" seguit de l'usuari. Aquesta informació en ser rebuda per el primer client serà tractada i mostrada a l'usuari que hi interactua.

1.3.9. Problemàtica 5: error en el continguts dels missatges

Tal com ja es comenta en l'apartat 1.2.2.5, es va haver de d'implementar la codificació en Base64 per tal d'assegurar la integritat de les dades que viatjaven dins de l'objecte XML.

Aquest codificació fa possible el traspàs d'informació entre sistemes sense pèrdua d'informació.

1.3.10. Problemàtica 6: funcions implementades no operatives

En el codi de projecte actual s'han implementat dues funcions corresponents a la creació de nous usuaris i nous grups que no estan operatives.

Aquestes dues funcionalitats corresponen a la crida de dues funcions de l'Api de LaCOLLA: addMember() i newGroup().

Les funcions han quedat implementades en espera de la seva modificació en el codi de LaCOLLA.

1.4. Valoració econòmica

Com ja s'ha indicat en apartats anteriors, tota la implementació d'aquest projecte s'ha realitzat utilitzant eines gratuïtes.

S'ha triat l'eina de desenvolupament NetBeans IDE 3.6 que es accessible a tot el món en sota llicència *SPL* (*Sun Public License*).

Com a llenguatge de programació s'ha utilitzat Java de Sun seguint els requeriments de l'enunciat. Aquesta plataforma esta subjecte a la llicència de *Sun Microsistems*, la qual no requereix pagament.

Per el que fa a les extensions utilitzades com ara són l'XML i Base64. El primer element està subjecte a la llicència *Apache Software License*.

L'implementació de la classe Base64 es publica com a *Open Source*.

Finalment, LaCOLLA també es considera una eina d'accés gratuït.

1.5. Conclusió

Aquest projecte planteja la solució a l'únic factor negatiu que tenen les aplicacions de missatgeria instantània que funcionen en l'actualitat a partir de la plataforma d'Internet.

El factor positiu recau en la descentralització del funcionament bàsic de l'aplicació.

En concret, aquest factor és el que facilita la infrastructura de LaCOLLA, següent aquest projecte una simple implementació sobre aquesta plataforma.

Per el contrari, existeix la problemàtica d'haver d'instal·lar tot l'entorn de LaCOLLA en cadascuna de les estacions d'usuari que desitgin participar en el grup.

2. Bibliografia

LaCOLLA: Develop a decentralized, autonomous and auto-organized infrastructure to facilitate collaboration and sharing of information among members of groups dispersed in Internet.
http://people.ac.upc.es/marques/LaCOLLA/

Java 2 Platform, Standard Edition, v 1.4.2 (J2SE)
http://java.sun.com/

NetBeans: free and open source IDE
http://www.netbeans.org/

RMI: Java Remote Method Invocation
http://java.sun.com/products/jdk/rmi/

Xerces: fully conforming XML Schema processor
http://xml.apache.org/xerces2-j/

Base64: public encoder/decoder
http://iharder.sourceforge.net/base64/

3. Annexos

3.1. LaCOLLA: Presentació

Document adjunt: *LaCOLLA_Article_no_acabat.pdf*

3.2. Api de LaCOLLA: Casos Dus i Contractes

Document adjunt: *Api de LaCOLLA Casos Dus i Contractes.doc*