
Il sistema Chat: esempio di documentazione

Autore:
Studente, Matr. xxxxxx

7 Dicembre 2006

Indice

1 Descrizione informale del problema.....	2
1.1 Glossario dei termini.....	2
2 Spazio del problema.....	4
2.1 Aspetti statici.....	4
2.2 Requisiti funzionali.....	5
2.2.1 Login utente.....	6
2.2.2 Logout utente.....	7
3 Spazio della soluzione.....	8
3.1 Architettura di sistema.....	8
3.1.1 Architettura Client.....	9
3.1.2 Architettura Server.....	10
3.2 Aspetti comportamentali.....	10
3.2.1 Login utente.....	10
3.2.2 Logout utente.....	12

1 Descrizione informale del problema

In questa sezione viene descritto in modo conciso il problema. E' richiesto di soffermarsi sulla descrizione (informale) dei principali requisiti forniti dal committente. Lunghezza suggerita 1-2 pagine

Obiettivo di questo lavoro è quello di sviluppare un sistema di dialogo multiutente gestito da un'unica unità centrale.

Il sistema è caratterizzato dalla presenza di tre entità logiche principali: l'utente, l'applicazione di interfacciamento (interfaccia client), e il gestore centralizzato degli accessi e della comunicazione (Server).

L'utente interagisce con l'interfaccia client impartendo una serie di comandi predefiniti e digitando messaggi da destinare ad altri utenti collegati. L'interfaccia client indirizza al server di comunicazione i comandi ricevuti dall'utente. Il server implementa i servizi richiesti ritornando eventuali comunicazioni al richiedente.

Gli utenti interagiscono con il sistema chat internamente a "stanze", compartimenti che compongono il mondo virtuale. Ogni stanza è caratterizzata da un nome ed una descrizione. Ogni personaggio può vedere o sentire solamente ciò che accade nella stanza in cui si trova. Nello specifico l'utente interagisce con la stanza attraverso l'interfaccia richiedendo servizi per la gestione della comunicazione con altri utenti connessi nella stessa stanza, attraverso i seguenti comandi:

- say <messaggio>, permette di inviare il messaggio a tutti gli utenti nella stanza.
- say_to <destinatario> <messaggio>, permette di inviare un messaggio al destinatario specificato presente nella stanza.
- emote <stato d'animo>, permette di comunicare lo stato d'animo indicato a tutti gli utenti nella stanza.

L'utente può inoltre interagire richiedendo servizi di gestione e monitoraggio di accesso al sistema e a specifiche stanze. Nello specifico si identificano i seguenti comandi:

- login <nomeUtente>, permette l'autenticazione di un utente e il conseguente accesso al sistema, l'utente connesso è automaticamente inserito in una stanza.
- logout, permette all'utente richiedente di abbandonare la stanza e il sistema.
- list_users, permette di visualizzare la lista degli utenti attualmente presenti nella stanza.
- look, permette di visualizzare la descrizione della stanza in cui si trova l'utente.
- map, permette di visualizzare la lista delle stanze disponibili.
- teleport <nome stanza>, permette all'utente di abbandonare la stanza corrente e trasferirsi nella stanza indicata.

....

....

1.1 Glossario dei termini

In questa sottosezione viene esplicitato il significato dei termini utilizzati nella descrizione presentata nella sezione precedente. E' richiesta l'individuazione dei principali stakeholders coinvolti e la definizione del ruolo svolto internamente al sistema.. Lunghezza suggerita 1 pagina

Utente – l'attore principale, è l'utilizzatore materiale del sistema. Si relaziona tramite l'interfaccia client al sistema di dialogo multiutente per comunicare con altri utenti connessi. Può assumere il ruolo di mittente o destinatario.

Sistema – il software che realizza l'applicazione. Basato su un'architettura distribuita. E' composto dall'interfaccia client e dal server.

Interfaccia client – Parte del sistema che gestisce la comunicazione con l'utente.

Server – Parte del sistema che gestisce la connessione e l'identificazione dell'utente.

Messaggio – mezzo con cui l'utente rappresenta e veicola l'informazione destinata ad altri utenti.

Emote – tipo di messaggio che l'utente usa per comunicare il proprio stato d'animo: può essere allegro, triste, arrabbiato.

Comando – mezzo con cui l'utente richiede un servizio al server.

Stanza – parte del sistema in cui gli utenti, identificati con una username, sono abilitati a mandare e ricevere messaggi tra di loro.

Chat – l'insieme delle stanze presenti nel sistema.

Username – identificativo univoco per ciascun utente della chat. Non deve contenere caratteri di controllo (spazio, tab, ritorno a capo).

Utente loggato – è un utente che ha accesso alla chat. Un utente loggato è sempre in una e una sola stanza.

Utente in una stanza – vedi utente loggato. Gli utenti in una stanza formano una partizione di utenti loggati.

....

....

2 Spazio del problema

In questa sezione si descrive lo spazio del problema precedentemente introdotto. E' richiesta la definizione dei principali requisiti e degli aspetti di struttura e di comportamento del sistema; in questa fase è necessario utilizzare il linguaggio UML e la metodologia dei casi d'uso.

2.1 Aspetti statici

Questa sottosezione deve mostrare il Diagramma delle Classi che definisce i principali aspetti strutturali del sistema specificando le principali entità coinvolte e le relazioni tra esse.

A livello dello spazio del problema, il sistema può essere modellato con il seguente diagramma.

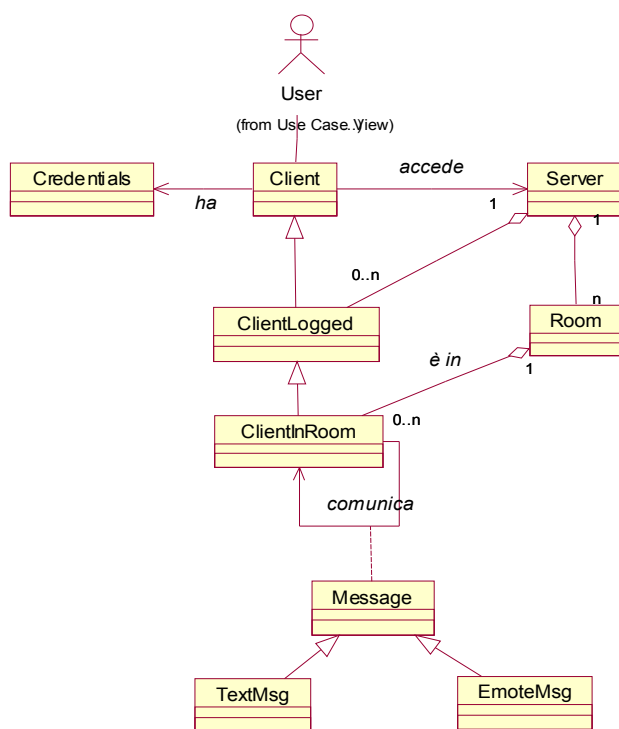


Figure 1 - Lo spazio del problema

L'utente rappresenta l'attore principale che interagisce con il sistema in vari scenari di utilizzo.

Il sistema è composto da una serie di moduli client e un modulo server.

Il client rappresenta l'applicazione che permette all'utente di usufruire di servizi offerti dal server e di mediare la comunicazione diretta tra vari utenti interfacciati.

L'utente attraverso il client interagisce con il server richiedendo il servizio di autenticazione. Un utente che ha effettuato un'autenticazione riuscita (LoggedUser) viene fatto entrare automaticamente in una stanza di default. L'utente in una stanza [UserInRoom] ha diritto di accesso agli altri servizi offerti dal sistema Chat.

Il servizio principale è quello di gestione di creazione e distribuzione di messaggi. Il messaggio rappresenta l'oggetto della comunicazione tra utenti, un messaggio è specializzato in messaggio di testo e messaggio emote.

Un utente in una stanza è in grado di comporre e spedire messaggi di testo a un particolare utente nella stessa stanza [Say_To] piuttosto che a tutti quelli nella stessa stanza [Say], è in grado di comporre e spedire

messaggi di emote [Emote], di listare gli utenti nella stanza [List], di chiedere una descrizione della stanza in cui si trova [Look], di chiedere di conoscere le stanza disponibili [Map], di cambiare stanza [Teleport] e di abbandonare il sistema[Logout].

....

....

2.2 Requisiti funzionali

Questa sottosezione deve mostrare un Diagramma dei casi d'uso che definisce in modo informale i principali requisiti funzionali richiesti, stabilendo le forme di interazione tra utente e sistema. Una opportuna descrizione dovrà commentare il suddetto diagramma. I vari requisiti introdotti nel diagramma devono essere ulteriormente dettagliati in forma testuale. E' richiesta la definizione di una sotto-sotto sezione per ogni caso.

Data l'individuazione dei singoli componenti che compongono l'architettura di sistema (Sez. 2) e la descrizione dell'architettura generale di alto livello dello spazio del problema, vengono qui presentati i casi d'uso che dettagliano i servizi offerti dal sistema Chat.

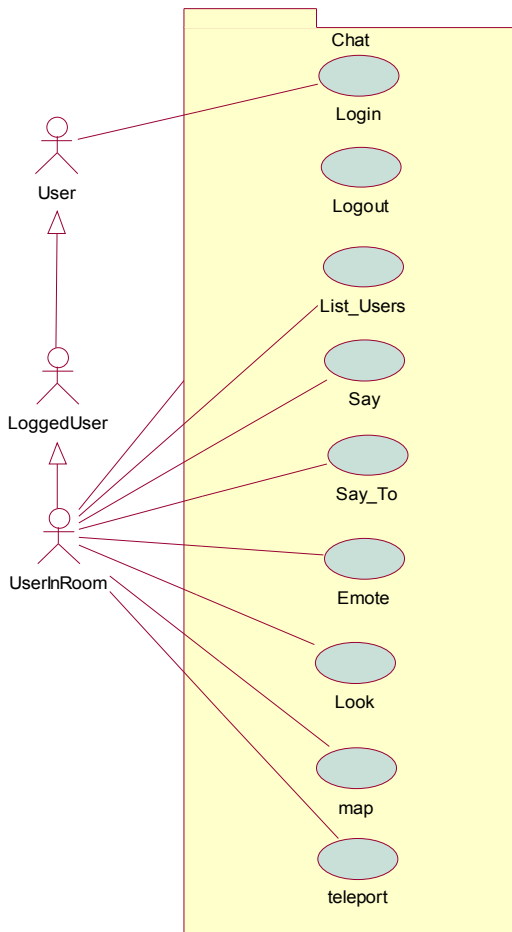


Figure 2 - Vista Funzionale

....

....

2.2.1 Login utente

Ogni caso descrive gli stakeholders coinvolti, l'attore principale, la descrizione dello scenario principale e di eventuali estensioni. Alla descrizione testuale deve essere affiancata una descrizione grafica che si avvarrà dell'utilizzo di Diagrammi di Interazione o di Diagrammi di Attività UML per definire (almeno) lo scenario principale del caso. E' naturalmente richiesto che i componenti coinvolti nella descrizione siano coerentemente conformi con quelli descritti nel Diagramma delle Classi mostrato nella sezione 2.1.

Primary actor: utente.

Stakeholders and interests:

utente – vuole accedere al sistema e da questo essere correttamente riconosciuto.

Precondition: l'utente è connesso al sistema.

Minimal guarantee:

Success guarantee: l'utente si logga nella chat e entra in una stanza di default.

Main success scenario:

1 l'utente invia il comando di login seguito dalla username . L'utente viene inserito nella chat (se non è già presente un utente con lo stesso nome) e entra in una stanza di default

Extensions:

1a Informazioni di login errate

1a.1 l'utente non viene autorizzato ad entrare nella chat: lo username proposto è già in uso.

1a.2 Il sistema deve dare la possibilità di un nuovo tentativo di accesso o di sconnettersi dal sistema.

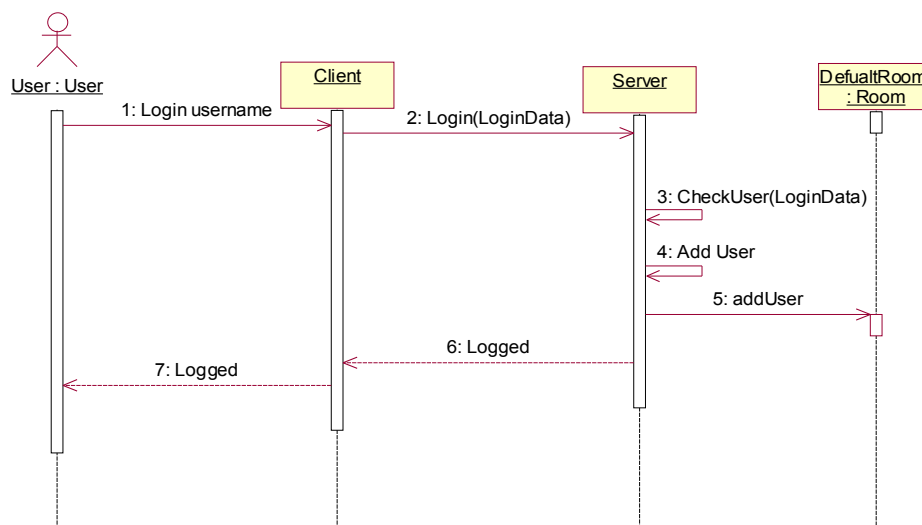


Figure 3 - Login effettuata con successo

L'utente invoca il comando login sul Client passando le proprie credenziali di autenticazione (userID). Il Client invoca il medesimo servizio sul Server. Il Server controlla la validità delle informazioni passate e ritorna un messaggio che certifica il successo o il fallimento dell'operazione. In caso di successo l'utente figura come loggato e può quindi usufruire di una serie di servizi offerti dal sistema Chat. L'utente loggato è automaticamente inserito in una stanza. In caso di fallimento l'utente può tentare nuovamente o abbandonare il sistema.

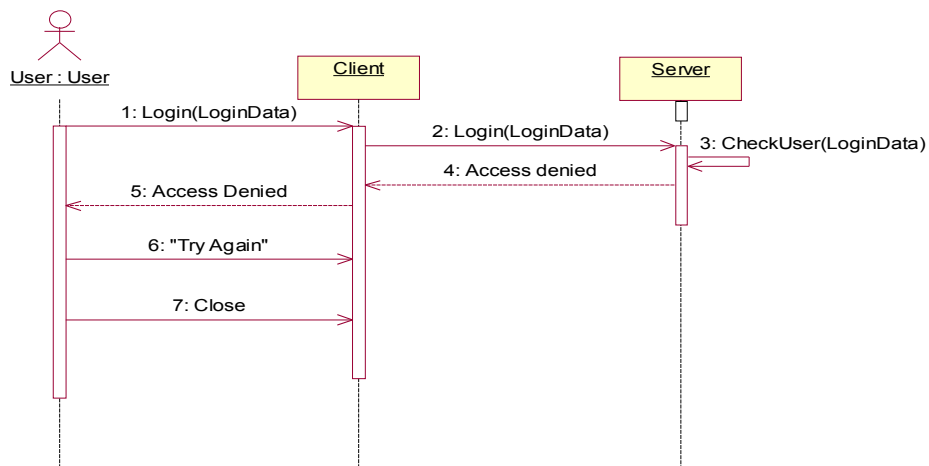


Figure 4 - Login fallito

2.2.2 Logout utente...

3 Spazio della soluzione

In questa sezione si descrive lo spazio della soluzione del problema precedentemente introdotto. E' obbligatoria la definizione dei soli aspetti strutturali, la definizione di aspetti di comportamento potrà dare luogo ad eventuale bonus in fase di valutazione.

3.1 Architettura di sistema

La sezione mostra un Diagramma delle Classi che descrive ad alto livello le relazioni tra i componenti dello spazio della soluzione. Un adeguato commento testuale deve motivare le scelte architetturali adottate. Varie sottosezioni descrivono più in dettaglio gli aspetti architetturali dei singoli componenti mostrati nel Diagramma delle classi.

Lo spazio della soluzione ad alto livello può essere modellato con il seguente diagramma.

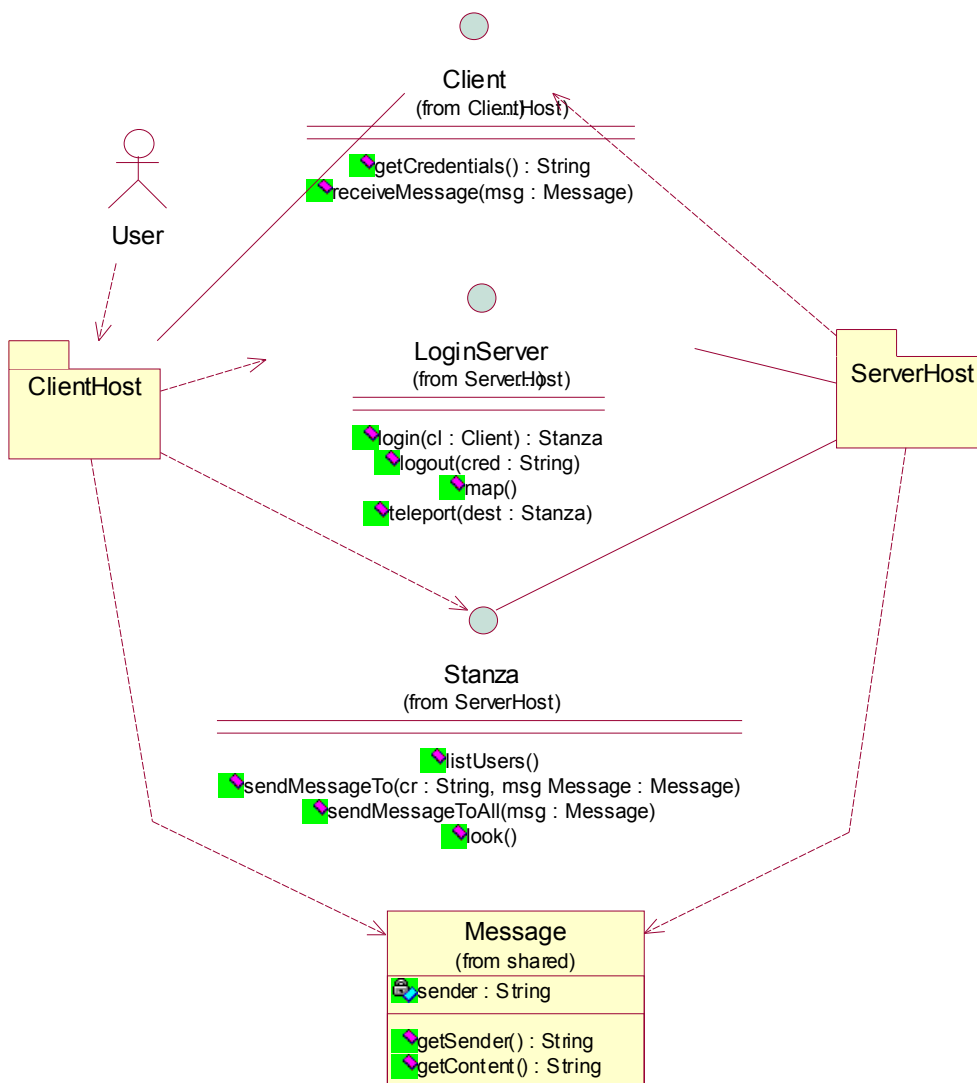


Figure 5 - Lo spazio della soluzione (alto livello)

L'utente rappresenta l'attore principale dei vari scenari di utilizzo e interagisce con il sistema attraverso un client il cui scopo è quello di mediare la comunicazione tra utente e server. Il Client dispone quindi di un'interfaccia verso l'utente e una verso il Server. In questo modo è in grado di mostrare ai rispettivi la visione dei servizi a loro dedicati.

Il Server è il cuore del sistema chat e opera mediando la comunicazione tra utenti. Il server è visibile all'utente attraverso due interfacce: LoginServer e Stanza. Tali interfacce specificano la visione del server in funzione del tipo di servizio che l'utente è abilitato a richiedere. In particolare LoginServer offre una visione di un servizio di autenticazione. Un utente viene autenticato attraverso la registrazione e il controllo (da parte del server) delle proprie credenziali (userID).

Un utente autenticato è in grado di usufruire degli altri servizi offerti (tramite l'interfaccia Stanza): può quindi richiedere la spedizione di messaggi (di testo o emote), di listare tutti gli utenti in quella stanza, di osservare la stanza in cui si trova, di muoversi in un'altra stanza o di abbandonare il sistema.

Client e Server condividono la struttura di definizione messaggio.

...
...

3.1.1 Architettura Client

In questa sottosezione è richiesto di descrivere in forma testuale e mediante l'utilizzo di Diagrammi delle Classi le caratteristiche statiche di un componente, evidenziando in particolare le scelte progettuali effettuate (riferendosi ad eventuali design patterns).

Dettagliando maggiormente lo spazio della soluzione il modulo Client del precedente modello ad alto livello viene modellato sfruttando il seguente diagramma delle classi.

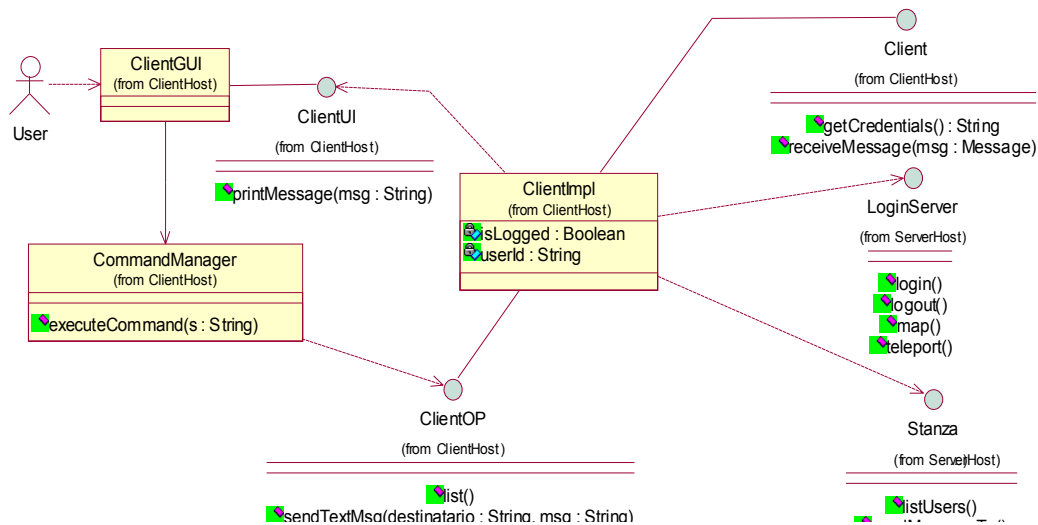


Figure 6 – Architettura del Client

Il modulo clientImpl implementa tutti i servizi richiesti all'applicazione client. ClientImpl offre un'interfaccia verso il server denominata Client e una verso l'utente denominata ClientOP. L'interfaccia ClientOP permette all'utente di invocare i servizi di autenticazione (login) e composizione e spedizione di messaggi.

L'interfaccia Client offre invece una visione dei servizi che il server è in grado di invocare. In particolare tale interfaccia definisce un metodo che permette al client di ricevere messaggi dal server.

La comunicazione con l'utente è gestita attraverso il modulo ClientGUI e dalla rispettiva interfaccia ClientUI.

ClientGUI gestisce una piccola interfaccia grafica che permette all'utente di comporre e inviare messaggi destinati al server o ad altri utenti e di visualizzare quelli ricevuti.

ClientGUI e ClientImpl interagiscono attraverso le rispettive interfacce ClientUI e ClientOP. ClientUI definisce un metodo che permette di appendere un nuovo messaggio alla lista di quelli precedentemente ricevuti e visualizzarlo. ClientOP offre una visione dei servizi resi dal client usufruibili attraverso l'interfaccia grafica.

....

....

3.1.2 Architettura Server...

3.2 Aspetti comportamentali

I casi d'uso introdotti a livello di spazio del problema vengono qui aggiornati ed estesi. E' richiesta la definizione di una sotto-sotto sezione per ogni caso.

Data l'individuazione dei singoli componenti che compongono l'architettura di sistema a livello di dettaglio e la descrizione dell'architettura generale dello spazio della soluzione, vengono qui presentati i casi d'uso che dettagliano i servizi offerti dal sistema Chat.

3.2.1 Login utente

Ogni caso deve descrivere gli stakeholders coinvolti, l'attore principale, la descrizione dello scenario principale ed eventuali estensioni. Alla descrizione testuale deve essere affiancata una descrizione grafica che si avvarrà dell'utilizzo di Diagrammi di Interazione o di Diagrammi di Attività UML per definire (almeno) lo scenario principale del caso. E' naturalmente richiesto che i componenti coinvolti nella descrizione siano coerenti con quelli definiti nei diagramma delle classi mostrati nella sezione 3.1 e 3.1.x

Primary actor: utente.

Stakeholders and interests:

utente – vuole accedere al sistema e da questo essere correttamente riconosciuto.

Precondition: l'utente è connesso al sistema.

Minimal guarantee:

Success guarantee: l'utente si logga nella chat e entra in una stanza di default.

Main success scenario:

1 l'utente invia il comando di login seguito dalla username . L'utente viene inserito nella chat (se non è già presente un utente con lo stesso nome) e entra in una stanza di default

Extensions:

1a Informazioni di login errate

1a.1 l'utente non viene autorizzato ad entrare nella chat. Causa: lo username proposto è già in uso.

1a.2 Il sistema deve dare la possibilità di un nuovo tentativo di accesso o di sconnettersi dal sistema.

Una descrizione di maggiore dettaglio è visibile attraverso i seguenti diagrammi di sequenza che modellano due cammini percorribili.

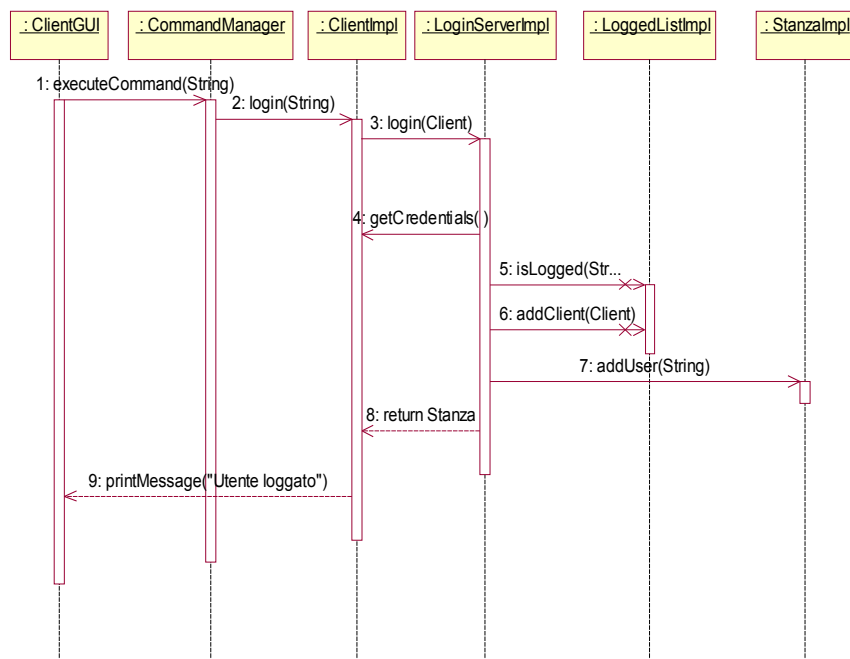


Figure 7 - Login effettuato con successo

L'utente digita il comando per effettuare la login "login *userid*" nell'apposita area di invio fornita dall'interfaccia grafica ClientGUI. Tale stringa viene opportunamente "parsata" dal CommandManager. Nel caso in cui venga riscontrato un errore nel formato del comando, viene mostrato un messaggio elencante i comandi possibili attraverso il metodo printMessage definito nell'interfaccia ClientUI e implementato dalla classe ClientGUI. Alternativamente se la stringa è riconosciuta sintatticamente corretta, viene invocata la funzione login di ClientOP implementata da ClientImpl.

ClientImpl invoca, a sua volta, il metodo login fornito dall'interfaccia LoginServer e implementato dalla classe LoginServerImpl. Tale metodo si occupa di recuperare le credenziali dell'utente e verifica, attraverso il metodo isLogged della classe LoggedListImpl, se un altro utente è già loggato con lo stesso identificativo. Nel caso in cui la userid proposta risulta essere libera, il Client che ha effettuato la richiesta viene registrato nella lista dei client attualmente loggati nella chat, e l'utente è inserito nella lista dei presenti in una stanza e viene ritornato il riferimento di un oggetto di tipo Stanza che permetterà al client di usufruire dei servizi di invio e ricezione di messaggi. Un messaggio che comunica la riuscita della login viene visualizzato dall'interfaccia grafica.

Nel caso in cui la userid risulti essere non valida(Fig. 9), il server lancia un' eccezione di tipo NotValid e un messaggio di errore viene propagato all'indietro fino all'interfaccia grafica. A questo punto l'utente può scegliere di riprovare l'autenticazione o abbandonare il sistema.

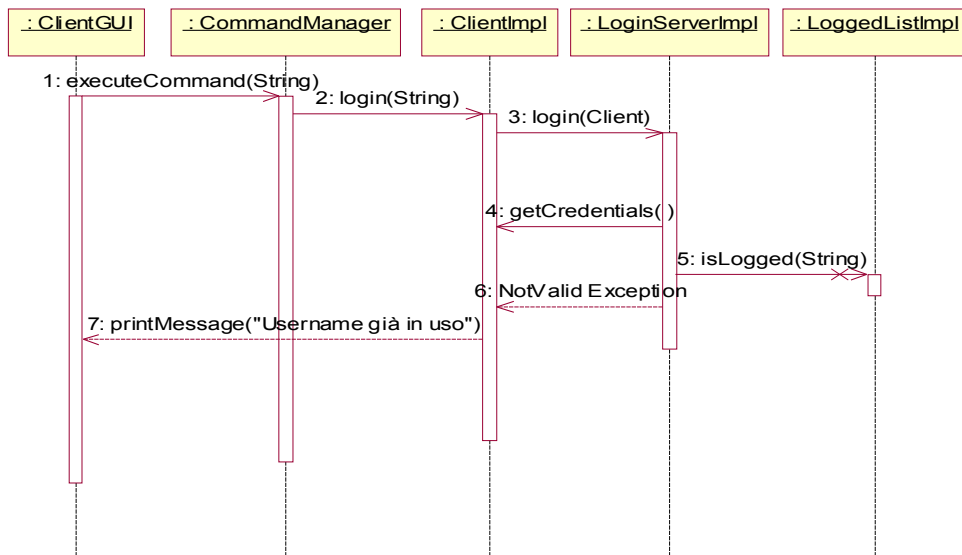


Figure 8 - Login fallito

3.2.2 Logout utente...