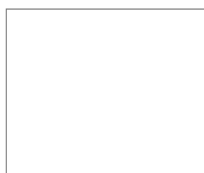


Pagina 1		ITIS Antonio Meucci Firenze	Rev 1.0 Data, 15/10/2020

Specifica dei Requisiti progetto Chat Server	Application Note

Data 15-10-20
 Versione 01
 Classificazione Public Release
 Nome del File SpecificaDeiRequisiti.doc

ITIS Antonio Meucci
 Via del Filarete 17
 50143, Firenze
 Italy

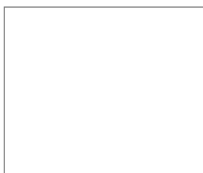


Descrizione Documento

Cliente	Scialpi, Benvenuti
Progetto	Chat Multi Utente
Titolo	Specifica dei Requisiti progetto Chat Multi Utente
Tipologia Documento	Application Note
Numero Documento	1
Versione	1
Data	15.10.20
Classificazione	Public Release
Autore(i)	Mathilde Patrissi
Approvato da	Mathilde Patrissi

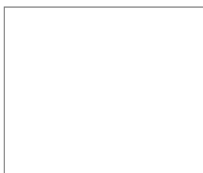
Approvazione Documento

Data	Nome	Titolo	Firma
15/10/2020	Mathilde Patrissi		



Indice dei Contenuti

<u>1</u>	<u>Introduzione.....</u>	<u>4</u>
1.1	Descrizione del documento	4
1.2	Descrizione del contesto	4
<u>2</u>	<u>Requisiti del progetto.....</u>	<u>4</u>
2.1	Requisiti funzionali del progetto	5
2.2	Requisiti non funzionali del progetto	5
<u>3</u>	<u>Ulteriori requisiti funzionali del prodotto.....</u>	<u>6</u>



.1 Introduzione

1.1 Descrizione del documento

Il documento è strutturato in tre capitoli il cui contenuto può essere riassunto come segue.

Nel primo capitolo (il presente) vengono riportate le informazioni di carattere generale che permettono di identificare il progetto a cui questo documento si applica.

Nel secondo capitolo vengono riportati i requisiti funzionali del progetto in questione.

Nel terzo capitolo vengono riportati ulteriori requisiti funzionali del progetto da implementare in una successiva fase di sviluppo.

1.2 Descrizione del contesto

Utilizzando il linguaggio di programmazione java è stata realizzata un'applicazione client-server che permette lo scambio di messaggi di testo tra due host.

All'avvio dell'applicazione server viene chiesto il numero di porta su cui deve ricevere i messaggi dai client.

All'avvio dell'applicazione client viene chiesto il nome del server, la porta su cui è in ascolto il server e a cui devono essere inviati i messaggi dai client e infine il nome del client.

.2 Requisiti del progetto

Lo sviluppo del progetto è stato affrontato prendendo come riferimento le informazioni e le richieste fornite dai professori, di seguito riportate:

Realizzare un server in grado di accettare le connessioni da 2 client:

- 1) il client dopo la connessione al server deve inviare come prima informazione il proprio nome*
- 2) il client, dopo aver inviato il nome, può inviare messaggi*
- 3) il server deve inoltrare i messaggi all'altro client*
- 4) il server deve gestire la situazione in cui ci sia solo un solo client collegato (impossibile inoltrare il messaggio)*
- 5) il client deve gestire attraverso un messaggio speciale la chiusura della connessione verso il server*
- 6) gestione della disconnessione del client*



- 7) i client devono conoscere i nomi dei client connessi
- 8) possibilità di avviare chat one to one oppure one to all

2.1 Requisiti funzionali del progetto

Inserimento dati e messaggi di informazione

All'avvio dell'applicazione server viene chiesto il numero di porta su cui deve ricevere i messaggi dai client.

All'avvio dell'applicazione client viene chiesto il nome del server, la porta su cui è in ascolto il server e a cui devono essere inviati i messaggi dai client e infine il nome del client.

Per entrambi è possibile utilizzare i parametri a riga di comando invece di utilizzare la modalità interattiva.

Caratteristiche della chat

Tra i client è possibile solo lo scambio di messaggi di testo.

Il server utilizza il meccanismo dei Thread per la gestione e l'invio dei messaggi, quindi ogni volta che un client effettua una connessione al server viene creato un thread. Questo si occuperà della comunicazione con il client in modo da permettere al server di gestire altre richieste di connessione. La comunicazione è gestita mediante l'utilizzo di un particolare protocollo che permette l'identificazione e la distinzione dei messaggi da inviare sulla chat (preceduti dal comando DATA_CMD) e dei comandi speciali (HELLO_CMD, QUIT_CMD, SETDEST_CMD, LIST_CMD).

Dalla cli vi è quindi la possibilità di utilizzare alcuni comandi come mostrato nello SplashScreen iniziale:

- 1) "q!" per lasciare la chat.
- 2) "dest:" seguito dal nome del destinatario, per comunicare unicamente con tale utente, oppure da "all" per comunicare con tutti gli utenti connessi
- 3) "list" per conoscere il nome degli utenti connessi alla chat

Tutto ciò è possibile anche mediante l'utilizzo dell'interfaccia grafica, gestita tramite swing. In questa modalità la lista, chiamata Rubrica, degli utenti connessi è presente nella list box situata in alto a sinistra, come mostrato dall'immagine seguente:

Cliccando sul nome dell'utente desiderato verrà effettuato l'inoltro all'opportuno destinatario.

Ogni volta che viene cambiato il destinatario vengono mostrati i messaggi scambiati con quest'ultimo. Ciò è possibile mediante l'utilizzo di un database sqlite dove vengono inseriti e letti i



messaggi nei momenti opportuni. Le comunicazioni da parte del server, ad esempio in caso di errori, vengono mostrate nella parte inferiore della chat. Vi è infine la possibilità di attivare e disattivare la musica di sottofondo e di settare la lingua desiderata.

2.2 Requisiti non funzionali del progetto

Requisiti di prodotto

Il progetto deve essere realizzato tramite il linguaggio di programmazione java e tramite il meccanismo dei thread .

Requisiti di consegna

Il progetto, che deve essere terminato entro il 9 Novembre, deve comprendere i file sorgenti, con gli opportuni commenti, e i javadoc.

Il progetto deve essere inserito in un repository git il cui url dovrà essere consegnato entro e non oltre la data suddetta.

.3 Ulteriori requisiti funzionali del prodotto

In una fase successiva del progetto si potrebbe ipotizzare la gestione di gruppi differenti e l'invito o la rimozione di utenti all'interno di tali gruppi.

Si potrebbe anche ipotizzare la gestione dello scambio di file multimediali, l'impostazione di una propria foto profilo e stato.

Infine si potrebbe implementare l'autenticazione dell'utente e la crittografia del canale di comunicazione.