

Programmazione di sistema

Anno accademico 2015-2016

Progetto del corso

Obiettivo del progetto

Sviluppo di una soluzione client-server per il controllo remoto di uno o più computer Windows.

Il sistema deve essere composto da due parti: una parte server, implementata in C++, e una parte client, implementata con qualsiasi linguaggio si voglia.

Parte server

Deve esporre un socket in attesa su una porta definita, al quale il client remoto si possa connettere. È sufficiente che il server gestisca un unico client alla volta. Non è richiesta la sicurezza della comunicazione.

Il server:

- deve determinare la lista delle applicazioni in esecuzione dotate di interfaccia grafica (attraverso la funzione di sistema EnumWindows) e dei relativi processi
- per ogni applicazione in esecuzione deve trovare l'icona associata
- deve conoscere quale di queste applicazioni dispone attualmente del focus
- deve poter mandare all'applicazione attualmente in focus combinazioni di tasti, composte da zero o più modificatori (CTRL / ALT / SHIFT) seguiti dal keycode corrispondente (a / b / c / ... / 1 / 2 / 3 / ... / backspace / delete / esc / ...)

Il server deve inizialmente inviare al client la lista delle applicazioni in esecuzione e, successivamente, comunicare gli eventi significativi (cambio di focus, chiusura di una applicazione, avvio di una nuova applicazione).

Il server potrà essere un'applicazione senza interfaccia grafica, o con un'interfaccia grafica minimale (eventualmente ospitata nella tray area).

Parte client

Il client è un'applicazione dotata di interfaccia grafica in esecuzione su una macchina diversa dal server. In seguito alla connessione col server, il client:

- ottiene la lista delle applicazioni in esecuzione, con il nome completo del processo e la relativa icona (se disponibile)
- riceve una notifica quando la lista od il focus cambiano
- visualizza un riassunto grafico dell'attività in corso sul server, mostrando le applicazioni attive, con relativa icona, (evidenziando quella dotata di focus) e la percentuale di tempo (da quando il client si è connesso) in cui ciascuna di esse è stata dotata di focus
- permette di inviare all'applicazione attualmente in focus una combinazione di tasti
- **Facoltativamente**, il client può connettersi a più server, mostrando le relative statistiche e permettendo di inviare combinazioni di tasti a tutti quei server su cui è in esecuzione una data applicazione.

SERVER

Thread 1

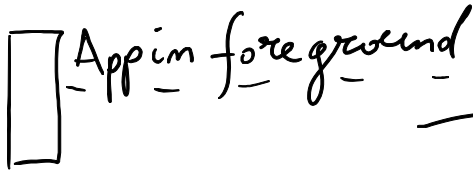
Aggiornamento
lista



MODIFICHE



Invio
modifiche



Invio
comandi

Thread 2

Ricezione
comandi



App

Thread 1

- * Creazione del socket
- * (in un while)
- * Attesa di connessione sul socket e creazione secondo thread
- * Creazione del list manager e passaggio del socket al list manager
- * Attivazione del ciclo principale del list manager (aggiornamento lista)
- * Dopo il while, pulizia e chiusura dell'applicazione

Thread 2

- * (in un while)
- * Attesa di comandi sul socket
- * Verifica del comando:
 - Comando per il server: eseguire il comando
 - Comando per l'applicazione in foreground: inviare il comando all'applicazione