

Commenti Rete GC01

Andranno implementate due architetture di reti differenti, per il momento abbiamo ragionato solo su RMI ma abbiamo comunque utilizzato delle interfacce per la futura implementazione di Socket. (I consigli sul come sono ben accettati)

L'interfaccia VirtualView sul lato Server e possiede metodi di comunicazione Server-Client (quindi funzioni di update della view principalmente).

L'interfaccia VirtualServer sul lato Client e possiede metodi di comunicazione Client-Server (quindi funzioni di gestione room e gioco effettivo).

L'interfaccia Client è utilizzata semplicemente per essere utilizzata come mezzo comune di comunicazione sia all'implementazione RMI che Socket.

Ci è stato suggerito nei laboratori di rendere il protocollo RMI asincrono, quindi utilizziamo una coda per salvare le azioni da eseguire senza bloccare la comunicazione.

Componenti del protocollo RMI

CLIENT

L'RMIClient, implementazione dell'interfaccia VirtualView, si occupa di chiamare i metodi dell'interfaccia VirtualServer e contiene anche metodi di aggiornamento della view.

SERVER

L'RMIServer, implementazione dell'interfaccia VirtualServer, si occupa di effettuare le chiamate ai metodi del controller e di chiamare metodi per mostrare le modifiche del model.

ACTION

Per salvare le azioni in coda abbiamo creato delle classi per i differenti tipi di comandi.

Sequenza di comunicazione del protocollo RMI

CLIENT → SERVER

Gli eventi generati dalla view chiameranno i metodi sull'interfaccia Client, implementata da RMIClient.

Verrà chiamato così il metodo sul VirtualServer, implementato da RMIServer, che crea l'azione e la inserisce in coda.

Successivamente, l'azione sarà estratta dalla coda e eseguita, chiamando il metodo sul rispettivo Controller.

Il Controller modifica il model.

SERVER → CLIENT

Il model si occuperà di notificare le modifiche a tutti i Client registrati ad esso, chiamando metodi sulla VirtualView, implementata dall'RMIClient.

RMIClient si occuperà di modificare la view con gli aggiornamenti ricevuti.