

PROGETTO DI RETI INFORMATICHE

635246 - PALA SEBASTIANO

INTRODUZIONE

L'applicazione consiste in un server centrale che permette ai client che vi si connettono di entrare in un Escape Room, e questi dovranno poi raccogliere tutti i gettoni all'interno della stanza scelta entro il tempo limite per poter vincere.

Le stanze sono implementate tramite una struttura dati **game_info**, che contiene tutte le informazioni necessarie per la corretta inizializzazione di un'istanza di gioco, rappresentata da una struttura dati **game_room**. Le varie strutture di tipo **game_info** vengono inizializzate all'avvio del server, con la lettura dei file di tipo "roomN.txt", che contengono lo stato iniziale degli oggetti, e i vari messaggi che vanno inviati come risposta agli utenti che fanno una partita nella stanza di tipo N. Di conseguenza l'aggiunta di nuove stanze è molto semplice, in quanto basta compilare un file di testo con il formato che il server può interpretare. Il server e il client comunicano tramite socket bloccanti utilizzando il protocollo TCP, in quanto l'applicazione è di tipo loss-intolerant. Ogni messaggio scambiato tra client e server presenta un header di dimensione fissa, che contiene informazioni sul mittente, sulla lunghezza e tipo di messaggio che si vuole inviare. A seconda del tipo del messaggio, dopo l'header verrà inviata anche una stringa di caratteri. L'header viene quindi inviato in modalità binary, mentre invece la stringa di caratteri in modalità text. L'opzionalità della stringa di caratteri all'interno del messaggio è vantaggiosa in quanto ci permette di scambiare informazioni inviando solamente l'header (che presenta una dimensione molto ridotta), riducendo dunque il carico sulla rete anche con un numero elevato di messaggi.

SERVER

Il server, oltre a gestire i messaggi degli utenti, è in grado di eseguire dei comandi ricevuti tramite input da tastiera. Oltre al comando di "stop", che arresta il server, sono stati aggiunti dei comandi aggiuntivi: "online" mostra a video la lista di utenti che ha effettuato il login, "rooms" mostra lo stato delle partite in corso, e infine "dc <id>" che permette di disconnettere un utente. Per poter fare le due cose contemporaneamente, è stato utilizzato l'I/O multiplexing. Il server inoltre è iterativo, quindi gestisce le richieste una per volta. Questo significa che per un numero elevato di richieste il tempo di risposta potrebbe aumentare di un quantitativo non trascurabile, ma perlomeno si evita l'overhead dovuto al cambio di processo.

CLIENT

Anche per il client è stato utilizzato l'I/O multiplexing, in modo da permettere all'utente di digitare comandi di gioco e ricevere i messaggi dal server contemporaneamente. Il client si comporta da interfaccia verso il server, in quanto non ha informazioni sulle stanze. Le uniche informazioni in suo possesso sono la sintassi dei comandi, quindi ha come compito quello di bloccare i comandi digitati in maniera errata e inviare al server solo i comandi corretti, in modo da ridurre il carico di lavoro del server e anche il numero di trasferimenti. Il client tiene traccia del proprio stato, in modo da sapere in qualsiasi momento quali sono i comandi che possono essere effettivamente utilizzati. Dopo essersi connesso al server, il client presenta una schermata di login, in cui è richiesto fare l'accesso per giocare. Dopo aver fatto l'accesso, è possibile uscire dall'account utilizzando il comando "logout".

Le informazioni dopo un eventuale signup verranno salvate dal server sul file shadow.txt.

PARTITA

Sul server è presente una sola stanza, caricata da room1.txt. In questo file sono presenti le informazioni necessarie all'inizializzazione della stanza. Il file contiene il nome della stanza da mostrare nel menu, lo stato iniziale degli oggetti, la descrizione da bloccato, enigmi ecc. Nonostante si parli di oggetti, il formato è utilizzato sia dagli oggetti che dalle locazioni. Infatti sono trattati allo stesso modo, e anche queste ultime possono avere una descrizione diversa a seconda dello stato in cui si trovano. Per evitare che si possano raccogliere le locazioni, si utilizza il separatore ' | ' all'interno del file per distinguere gli oggetti che possono o non possono essere raccolti. Questo approccio è stato scelto per garantire la massima flessibilità nella creazione di stanze future, permettendo di scegliere il numero di oggetti e locazioni a piacimento, finché non si supera il limite massimo di oggetti definito nel file costanti.h. Una volta che un giocatore entra in una stanza, il server crea un'istanza privata di quest'ultima utilizzando dei puntatori alle aree di memoria contenenti i messaggi di testo della stanza corrispondente, e creando una copia privata dello stato iniziale degli oggetti. Per vincere è necessario raccogliere tutti gli oggetti con stato iniziale SPECIAL_LOCKED entro il tempo limite. Sono stati aggiunti 2 comandi: "drop <slot>" rimette a posto un oggetto dell'inventario, e "leave" che permette all'utente di uscire dalla stanza e tornare al menu. La funzionalità a scelta implementata è la possibilità di fare una partita a 2 giocatori, utilizzando "coop <room>". Se solo un giocatore ha eseguito questo comando, rimarrà in attesa che un altro giocatore esegua il medesimo comando sulla stessa stanza. Il client tiene traccia di che tipo di partita si sta giocando, e se è a 2 giocatori permette l'utilizzo di 2 comandi aggiuntivi: "give <slot>", che dona un oggetto all'altro giocatore, e "say <mess>" che invia un messaggio all'altro giocatore. Un giocatore può uscire dalla stanza in qualsiasi momento, e questa situazione è gestita dal server ripristinando gli oggetti nell'inventario del quitter, e notificando il giocatore del passaggio ad una partita single player, rimuovendo la possibilità di utilizzare i comandi "give" e "say".

SOLUZIONE

Gettone rosso: [use interruttore, look monitor (opzionali, per il codice)] take gemma,118, take gemma, use gemma statua, take chiave, use chiave cassaforte, take gettone rosso

Gettone blu: take accendino, il vizio, take accendino, use accendino corda, take martello, use martello vetrina, take lancia, use lancia gettone blu, take gettone blu

Gettone giallo: take gettone giallo,2, take gettone giallo