

PEER-REVIEW 2: COMMUNICATION PROTOCOL

Chiara Angileri, Nicholas Beltramini, Stefano Arcaro
Gruppo AM06

5 Maggio 2022

Valutazione del protocollo di comunicazione del gruppo AM36

1. Lati positivi

- In generale il protocollo di comunicazione è ben progettato, inoltre il documento è molto esaustivo.
- Serializzazione con JSON: riteniamo che la decisione di optare per una serializzazione tramite JSON piuttosto che implementando l'interfaccia *Serializable* di Java sia ottima, per due motivi:
 - Una serializzazione *language-agnostic* permetterebbe un'eventuale comunicazione tra client e server scritti anche in linguaggi diversi.
 - Rende più semplice il debugging nel caso in cui fosse necessario.
- Ci piace la soluzione adottata nel caso in cui il server cada e al ripristino i giocatori possano riconnettersi e riprendere la partita.
- Condividiamo la scelta di suddividere la comunicazione in diverse fasi in modo da poter determinare con maggiore facilità la coerenza di un messaggio ricevuto rispetto alla fase di gioco.

2. Lati negativi

- Riteniamo che possa essere conveniente avere un messaggio che informi i giocatori riguardo gli effetti dei personaggi estratti qualora la modalità di gioco fosse esperta.

3. Confronto tra i protocolli di comunicazione

- Anche noi abbiamo optato per serializzare tramite JSON in modo da sfruttare i vantaggi precedentemente elencati, oltre che per imparare a lavorare con un linguaggio di descrizione formale molto utilizzato.
- Abbiamo inoltre scelto, come voi, di aggiornare i client tramite messaggi specifici relativi alle avvenute modifiche durante la fase di gioco e salvando quindi anche a lato client una copia del modello.
- Abbiamo scelto, come voi, di implementare la funzionalità avanzata delle partite multiple gestendo però la fase di lobby diversamente: ogni giocatore che si connette specifica il nickname e le proprie preferenze riguardo al numero di giocatori e alla modalità di gioco. Se esiste già un gioco con quei settaggi non ancora iniziato il giocatore viene automaticamente aggiunto, altrimenti se ne crea uno nuovo.
- A differenza vostra noi abbiamo scelto di far partire il messaggio di PING dai client a cui il server risponde con un messaggio di PONG. Sia lato client che lato server è presente un timer che viene resettato alla ricezione di un messaggio e allo scadere del quale si suppone che la connessione (lato server o client) sia caduta e il programma termina di conseguenza.