

Peer-Review 2: Network

Samuele Scherini, Matteo Spreafico, Ludovica Tassini

Gruppo GC58

Valutazione della rete del gruppo GC03.

Lati positivi

Il diagramma UML della rete è completo e dettagliato, così come il materiale fornito dal gruppo per agevolarne la comprensione. La parte di comunicazione risulta ben organizzata, ed è anche stata implementata la classe `Logger`, che permette di mostrare in console messaggi formattati per avere un riscontro su ciò che sta accadendo “dietro le quinte”. Inoltre, per gestire l’eventualità in cui un giocatore perda la connessione, è stato introdotto un `ping system` lato server, il quale ogni 10 secondi invia un `CommunicationMessage` di tipo `ping`.

Lati negativi

Non sono stati riscontrati evidenti lati negativi. Come detto in precedenza, la rete risulta ben implementata.

Confronto fra le architetture

La progettazione della rete e le tipologie di messaggi del protocollo di comunicazione di entrambi i progetti sono simili. Il gruppo è molto avanti con il lavoro e dunque ha pensato anche a classi utili per le funzionalità aggiuntive. Sebbene le medesime funzionalità saranno di interesse anche per noi, le nostre corrispettive sono ancora in uno stato embrionale poiché il progetto base non è ancora ultimato (mancando la `view`).

Il gruppo ha deciso di implementare una sola classe, `CommunicationMessage`, per i messaggi scambiati fra Client e Server. In base ad un ID (di tipo `MessageType`) si può sapere se il contenuto contiene un `update` grafico della board, azioni richieste dal client ed altro ancora. Il nostro gruppo invece, ha preferito creare una classe per ogni tipo di messaggio scambiato e questo ha comportato implementare una classe madre `Message` e i relativi figli (tutti i possibili tipi di messaggio scambiati fra Client e Server), che ereditano da essa. Così facendo nel diagramma UML sono state segnate una decina di classi, per ora semi-vuote, che vanno ad appesantire il grafico, mentre la scelta di avere un solo tipo di messaggio semplifica molto il diagramma UML. Tuttavia, avere il tipo del parametro fisso per ogni tipo di messaggio può rivelarsi utile per un controllo preliminare della correttezza del valore inserito.