

# Peer-Review 2

<Matteo Briscini>, <Luca castelli>, <Riccardo Caprioglio>, <Davide Arcaini>

Gruppo <AM19>

Valutazione del data diagram delle classi del gruppo<AM28>.

## Lati positivi

La scelta di inviare messaggi interamente con l'ausilio di json permette di rendere la logica facilmente convertibile in socket o RMI.

La possibilità di avere un messaggio redo per rendere reversibile tutte le mosse del client ci è parsa una buona idea.

## Lati negativi

Non siamo stati in grado di capire come il server aggiorna bookshelf e main board sui client durante le fasi di partita dopo che viene pescata una o più carte.

Per specificare il numero di giocatori si occupa il server di richiedere al client il dato, in logica reactive e non proactive; per evitare questo approccio sarebbe bastato un parametro opzionale nel metodo per creare una partita, prendendo quindi le partite a 4 giocatori come settaggio di default.

Si poteva evitare un l'utilizzo di un messaggio per riordinare le tiles assumendo che il client comunichi al server un array di titles già ordinato, inoltre così facendo l'invio delle tiles e la selezione della colonna di inserimento potevano esser strutturati in un unico messaggio.

## Confronto tra le architetture

L'assenza del multi-partita semplifica notevolmente il compito progettuale per il protocollo di comunicazione client server, tutta via, al netto di errori marginali, nel complesso quello presentato è un protocollo solido.