

# Peer-Review 2: Rete

Mirko Pica, Riccardo Pianalto, Christian Prendin, Aldo Pinelli

Gruppo GC01

Valutazione del protocollo di rete del gruppo GC48 (sia class che sequence diagram).

## Lati positivi

- Buon utilizzo per l'interfaccia common client comune per socket ed rmi
- L'utilizzo della classe astratta Action garantisce una buona estendibilità del codice, inoltre aiuta correttamente a rendere asincrono rmi
- UML molto chiaro e leggibile
- La divisione tra ControllerManager e Controller permette di gestire al meglio le partite multiple, poichè ogni controller può essere eseguito nel proprio thread
- Corretto utilizzo dell'Observer manager per notificare le azioni sul server
- Sequence diagram chiari ed efficaci

## Lati negativi

- Potrebbe essere una buona idea aggiungere un nickname anche alla classe SocketClient per poter differenziare i vari clients
- Nelle actions, il ControllerManager potrebbe essere passato come parametro del metodo execute, piuttosto che come attributo. In questo modo, il client potrà chiamare direttamente le azioni, anzichè chiamare diversi metodi a seconda dell'azione desiderata.
- L'interfaccia Message dovrebbe essere una classe astratta, in quanto un'interfaccia non può avere attributi privati

## Confronto tra le architetture

- Le architetture sono molto simili, anche noi abbiamo creato una classe analoga alla vostra classe Action per rendere incapsulare i messaggi in una coda presente sull'RMIServer e rendere asincrone le chiamate.
- Anche noi abbiamo usato una CommonClient interface per avere un'interfaccia comune tra socket ed rmi
- Il protocollo di comunicazione per la parte RMI è pressochè identico al nostro