

SPIEGAZIONE UML SERVER GC18

Premessa: Per tutte le classi del Model non presenti fare riferimento al precedente UML, tenendo a mente che la classe Game è stata divisa in Controller e Model.

- Quando un giocatore si collega, il Server accetta la connessione e crea una **SocketClientConnection**, che da quel momento riceve i messaggi dal Client (sia in fase di Inizializzazione sia durante la partita), e una **RemoteView**, entrambe associate a quel giocatore/Client.
- MessageReceiver è una classe interna a RemoteView che osserva la SocketClientConnection.
- Il Controller è osservatore di ciascuna delle View.
- Le RemoteView osservano il Model.
- Quando arriva un messaggio, la SocketClientConnection notifica la MessageReceiver che chiama la View, che a sua volta notifica il Controller (che esegue la mossa sul Model).
- Terminato ciò il Model notifica i cambiamenti (se stesso) alle RemoteView che li trasmettono a tutti i Client.
- Le mosse vengono passate dal client come stringa.
- Il model viene passato ai Client tramite serializzazione.

