

Ciao

Vi inviamo la nostra idea di comunicazione client-server e come i metodi vengono eseguiti nel nostro progetto.

Come allegato dovrebbero esserci gli UML della parte di rete (Client, RemoteView, Action, GameMessage, ecc.) e del controller + model (che ha subito qualche modifica dall'ultima volta) con anche i sequence diagram per la parte di connessione dei client e creazione della partita (Connection) e quello per l'esecuzione dei metodi (Game).

Per la comunicazione client-server:

- Il client ha una lista di oggetti GameMessage.
- Ogni GameMessage rappresenta in realtà un'azione che l'utente può scegliere di fare, identificata da un indice, e che verrà stampata sul terminale dell'utente per fargli effettuare la scelta.
- Ogni GameMessage ha al suo interno due metodi: GameMessage.getMessage(), chiamato una volta che l'utente ha inviato il codice del metodo che vuole eseguire, e che richiede all'utente tutti i parametri che servono al controller per eseguire il metodo richiesto e li salva al suo interno (numero di studenti, colore degli studenti, ecc.) effettuando un controllo sull'input ricevuto per ogni parametro, e GameMessage.handleMove() che crea un oggetto Action che erediterà da ActionEvent, passando se stesso come attributo (spiegato meglio dopo).
- GameMessage verrà quindi serializzato e inviato alla RemoteView del client che chiamerà GameMessage.handleMove().

Per l'esecuzione dei metodi:

- Alla ricezione di GameMessage serializzato, la RemoteView eseguirà il metodo RemoteView.handleMove() al cui interno viene eseguito GameMessage.handleMove() con cui verrà creato un oggetto Action diverso per ogni GameMessage, contenente GameMessage (e quindi i parametri inviati dall'utente).
- L'oggetto Action sarà inviato alla actionPerformed del controller Game_Manager_Normal/Expert in ascolto in quanto listener, grazie al fatto che Action eredita da ActionEvent.
- Game_Manager_Normal eseguirà quindi il metodo execute dell'oggetto Action ricevuto che chiamerà il metodo del controller voluto andando a modificare il Model.
- Una volta terminato il metodo da eseguire il Model notificherà la RemoteView con relativa stampa sul terminale del client della situazione generale del gioco.