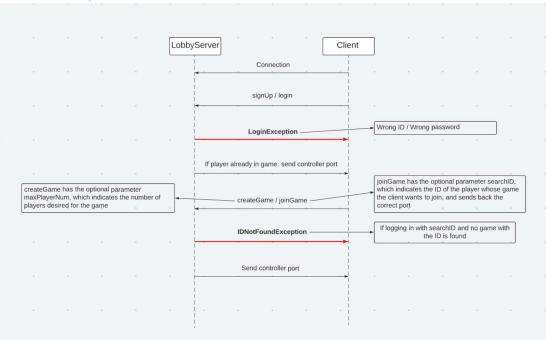
COMMUNICATION PROTOCOL

Introduzione

Vista la scelta di implementare la funzionalità avanzata delle partite multiple, abbiamo deciso di suddividere il protocollo di comunicazione in base allo stato del player (in lobby o in game), realizzando quindi due classi RMI e due per il protocollo TCP.

Per evitare di tener un riferimento unico che permetta di reindirizzare le mosse ricevute dai client alla partita corretta, si è scelto di assegnare una o due porte diverse per ogni partita (a seconda del numero di protocolli richiesti dai client) sono invece sempre aperte due porte alla quale i client possono raggiungere la lobby, che si occuperà di allocare e comunicare i client le porte a cui raggiungere le partite

Fase in Lobby



la lobby si occupa di autenticare i client, creare le partite e permettere il collegamento a queste.

In fase di login si verifica se il client è già registrato come giocante in una partita, in questo caso si considera come fosse avvenuta una disconnessione indesiderata e si ricollega il client alla partita.

altrimenti l'utente può:

- Creare una partita, specificando opzionalmente il numero di player massimi
- Entrare nella prima partita libera disponibile
- Entrare in una partita con l'id di un player partecipante

In ingresso ad a una partita la lobby comunica al client la porta alla quale raggiungerla

Sono riportati due casi d'errore con relative eccezioni se i dati di login sono errati o se in fase di join to the game l'id giocatore specificato non viene trovato

Fase in game Server Client pre game sendPlayersNumbe startGame if the game is just started, there is messaggi enough player or the players hasn't condizionali the privileges to start the game sendPlayersList sendAllPlayerBoard sendMainBoard sendAlllCommonGoal sendPrivateGoall each players in game notifyActivePlayer all control are duplicate on server and on 1. taking to much or to less card not frough space on the player board not free space near the card is the turn of an other player to all the clients, so it will trigger the two case under if necessary addCardToClientBoard dellCardFromMainBoard end game sendEndGamePoint in each case is always possible to a client to reconnect to the game connection manager

Suddividiamo lo svolgimento della partita in tre fasi:

- ingresso e start del game con conseguente creazione dei dati dei client
- svolgimento del gioco: in questa fase il server comunica al client che è il suo turno, il client potrà quindi effettuare la mossa
- end game: in questa fase il server calcola il punteggio finale della partita e stabilisce il vincitore comunicandoli a tutti i client

in parallelo è tutte e tre le fasi di game sono forniti al client i metodi per gestire lo stato di connessione (connection e quit connection) e un sistema di ping-pong per gestire disconnessioni impreviste