

Mazzo da 40

Cuori — Quadri — Picche — Fiori
123467JQK

Porta di comunicazione 8889

HOST = chi manda la richiesta
CLIENT = chi accetta la richiesta

4 carte le metti giù scoperte e ne consegni 3 a testa
Se carta dello stesso numero la si prende se no la si aggiunge
Quelle prese non vanno coperte
Se ho la stessa carta che ha sopra al mazzo l'avversario la puoi prendere
Vince chi ha più carte

(1)

Inserisci il nickname - ip altro utente - aspetta che accetta c

HOST => `con;nickname;`

CLIENT => `con;nickname`(del secondo utente);

(2)

Ognuno lancia un dado da 1 a 6 chi ha il numero più alto dei due parte per primo

Il CLIENT manda all'HOST il risultato

`dad;5;`

L'HOST confronta i due risultati e decide chi parte

`dad;client;`

`se dovesse partire l'host diventa dad;host; se uguali i risultati dad;retry;`

Nel caso di numeri uguali si riparte subito al punto (2)

CHI PERDE DIVENTA HOST PARTITA

(3)

- HOST mescola le carte

- HOST invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz;nMaz;`

- HOST invia `crd;cartaUser, cartaUser, cartaUser;`

es. 1C => asso di cuori QQ => donna di quadri

- una volta arrivate al CLIENT si parte

Situazione attuale:

PEER1=> nome utente avversario, 4 carte sul tavolo, 3 carte personali (diminuiscono)

refresh tavolo con i nuovi dati inviati dall'avversario se presenti

GIOCATA

-caso RUBA MAZZO

invia `rub;`

invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

-caso PRENDE CARTA

invia `pre;`

aggiornando i suoi dati

invia `tav;cartTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

-caso AGGIUNGO CARTA SUL TAVOLO

invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

PEER2=> nome utente avversario, 4 carte sul tavolo, 3 carte personali (diminuiscono)

refresh tavolo con i nuovi dati inviati dall'avversario se presenti

GIOCATA

-caso RUBA MAZZO

invia `rub;`

invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

-caso PRENDE CARTA

aggiornando i suoi dati

invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

-caso AGGIUNGO CARTA SUL TAVOLO

invia `tav;cartTav, cartaTav, cartaTav, cartaTav; maz(nome carta sopra); nMaz(numero carte); (obbligatorio)`

Quando finisce l'HOST da le carte ogni 3 turni

Quando l'HOST vede che non ci sono più carte nel mazzo = in base ai dati che ha determina il vincitore

Invia win;nickname;punteggioVincitore;punteggioPerdente;
(nickname del vincitore)

In base a chi prende per ultimo una carta dal tavolo le carte rimanenti saranno sue.

REGOLE SPICY:

- CONTEGGIO ORI: chi ha più ori nel mazzo è come se avesse 5 punti in più sul totale.
- CARTE INFINITE: non terminano mai le carte