User identify: IP address + User ID

Platform identify: WEB = 1, Android = 2, iOS = 3,(maybe there are others)

Thin server and fat client:

Server takes in charge of match a game, transmit message between opponents and end of a game.

Client takes in charge of judging legal steps and judging whether the user wins the game.

Architecture of Server routine:

Web clients’ server

Common Socket Layer

...

Android clients’ server

Common Data

--Common Socket Layer listen the connection request whatever the platform you use, by the initial exchanging message, then assign this connection to which server should be responsible.

--Common Data is the data used to maintain a match and vacancies, it should include: match list, vacancies list for waiting(there is a user is waiting in a match table), vacancies list is idle(there is no one in a match table).

Sever Receive Messages:

enum message\_receive = { START, WIN, REGRET\_REQUEST, REGRET\_OK, TURN\_OVER, DOG\_FALL}

enum message\_send = {END\_GAME, YOUR\_TURN, CAN\_REGRET, REJECT\_REGRET, REQUEST\_START}

Client should maintain a data structure named Board

Board

{

bool position[10][10];(member)

String opponentID;

Int boardID;

bool IsLegalStep(int x, int y);(method)

bool SetPostion(int x, int y);(method)

}

Communication

