Network Documentation

1 Introduction

The Client-Server communication protocol is based on string exchanges encoded following the JSON syntax.
The exchanged messages are JSON-Object having at least a section called "type" (string) that defines the message also known as EVENT

2 Server-Client

2.1 "type" : "heartbeat"

Used to check the state of the TCP connection (if the client is still up/online);

2.2 "type" : "notification", "msg" : "..."

Used to notify the client about the events (has no logic use);

2.3 "type" : "message", "to" : "receiver player", "from" : "sender player", "message" : "..."

Used for the chat feature, if the "type" : "to" is empty, will be a broadcast message;

2.4 "type" : "disconnected", "player" : "player name"

Used to warn other online players that a player has logged out;

2.5 "type" : "setUsername", "username" : "desired username"

Used to confirm to a player that the chosen username is available and can be used;

2.6   **"type" : "setup",**
"setup" : {PlayerSetup},
"publicGoals" : ["...", ...],
"playerNames" : [...],
"colors" : [...],
"normalDeck" : [...],
"goldenDeck" : [...]

- **"setup" : {PlayerSetup} :** class that contains the setup of a specific player;

- **"publicGoals" : [...] :** encoded strings of public goals;

- **"playerNames" : [...] :** strings of player names;

- **"colors" : [...] :** colors of each player following the same order of players;

- **"normalDeck" : [...] :** strings of the NormalDeck cards that can be drawn (max 3 strings!);

- **"goldenDeck" : [...] :** same as above.

2.7   **"type" : "turn",**
"player" : "player name",
"normalDeck" : [...],
"goldenDeck" : [...],
"scores" : [...],
"gameFinished" : " "

- **"type" : "turn" :** always send as a broadcast to update every player about the current turn;

- **"player" : "player name" :** name of the player that is starting the turn, if player = " " there are two options :

  1. game ended because you are the only player left;
  2. game ended successfully, indeed the specified scores contain the final scores of every player;

- **"normalDeck" : [...], "goldenDeck" : [...] :** same as above;

- **"scores" : [...] :** contains the scores of every player;

- **"gameFinished" : " " :** it flags that the game has come to an end.

3 Client-Server

3.1 "type" : "heartbeat", "type" : "message"

Same as above;

3.2 "type" : "login", "username" : "desired username"

The client asks the server if the username is available;

3.3 "type" : "numOfPlayers", "numOfPlayers" : int

The first player must send this to the server to let the server know how many players are going to join (min = 2, max = 4);

3.4 "type" : "setup", "starterCard" : "...", "goalCard" : "..."

Tells the server the choices made:

- **"starterCard" : "..." :** selected starter card (encoded);

- **"goalCard" : "..." :** selected goal card (encoded);

3.5 "type" : "turnActions", "placedCard" : {PlacedCard}, "deckType" : boolean, "indexVisibleCards" : int

- **"placedCard" : {PlacedCard} :** object that specifies the placed card and its position on board;

- **"deckType" : boolean :** specifies from which deck has drawn the player:
  -**false :** normalDeck;
  -**true :** goldenDeck;

- **"indexVisibleCards" : int :** specifies index of the drawn card (min = 1, max = 3, 3 means the flipped card)

3.6 "type" : "quit"

Notify the server that the player has logged out.