

Communication protocol

AM54

Color legend:

- Red messages: Server → Client
- Blue messages: Client → Server

{JSON}

General messages (found in multiple phases)

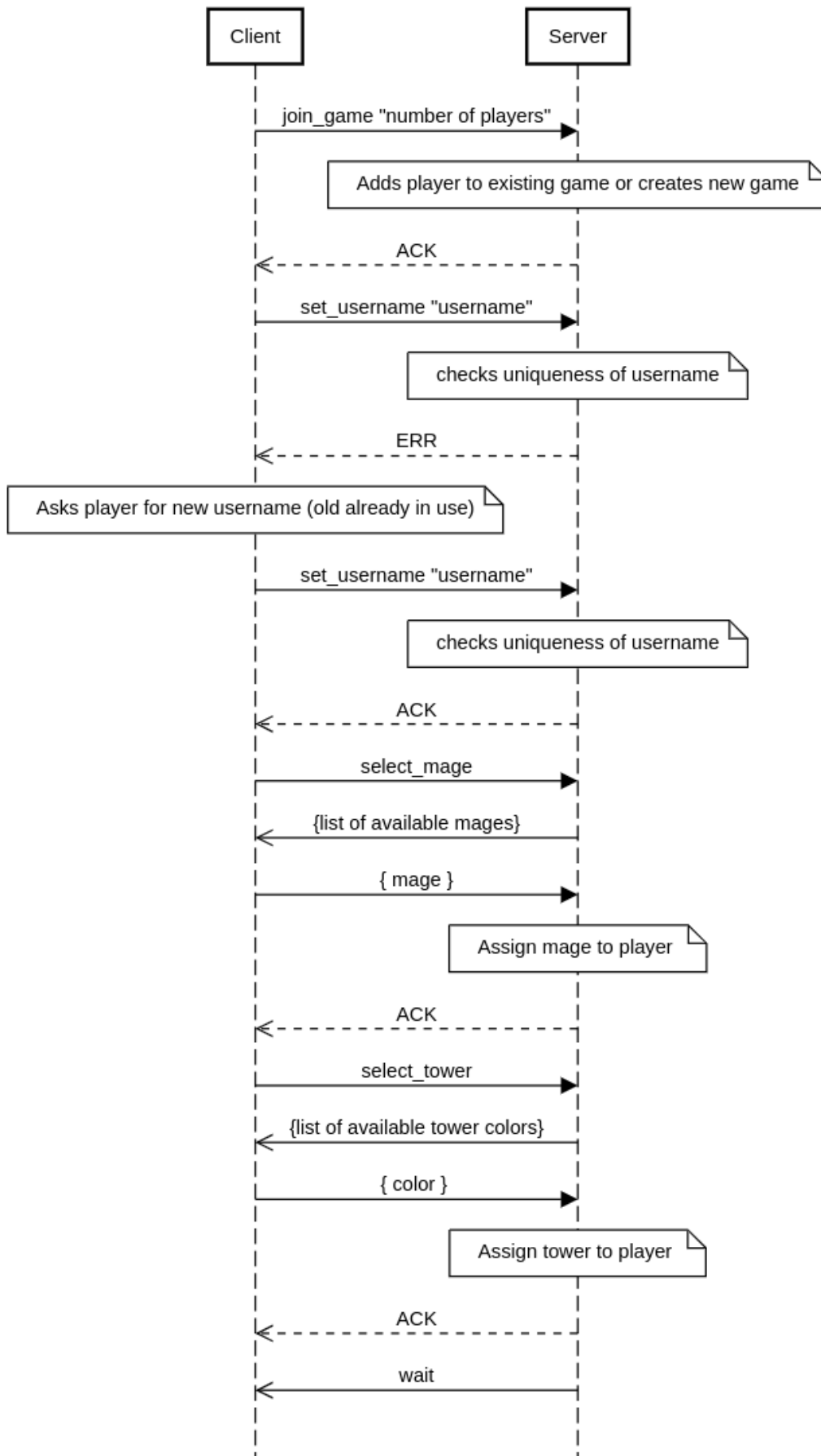
- **ping** - sent by Server to assure that Client is still connected, before every Server command
- **pong** - client's responses to ping
- **ERR** - sent as response by Server, indicates that there was error and usually Server expects new input
- **ACK** - sent as response by Server to assure that everything went as expected and no further actions are necessary (for preceding command sent by client)
- **wait** - indicates to client that it shouldn't send any communication to server until it receives `next_turn` or `select_assistant_card` command (it can only respond to ping)

Initial Phase

In this phase, after establishing a connection client will send messages to the server asking to set username and to get lists of available mages and colors of towers, from which he will choose desired mage and color. After the end of selection, the client will receive the message "wait" and will enter in the waiting state.

- **join_game "number of players"**
 - **join_game** - asks server to join game with certain number of players, if such game doesn't exist, server creates it
 - **"number of players"** - int that indicates desired number of players
- **set_username "username"**
 - **set_username** - notifies server that client wants to select username
 - **"username"** - string that contains username selected by player
- **select_mage** - notifies server that client wants to select a mage and asks for list of available mages
- **{list of available mages}** - list of not selected mages, from which player should select one
- **{mage}** - player selection of mage
- **select_tower** - notifies server that client wants to select a color of towers and asks for list of available colors
- **{list of available tower colors}** - server sends list of available colors from which client should select one
- **{color}** - player's selection of towers' color

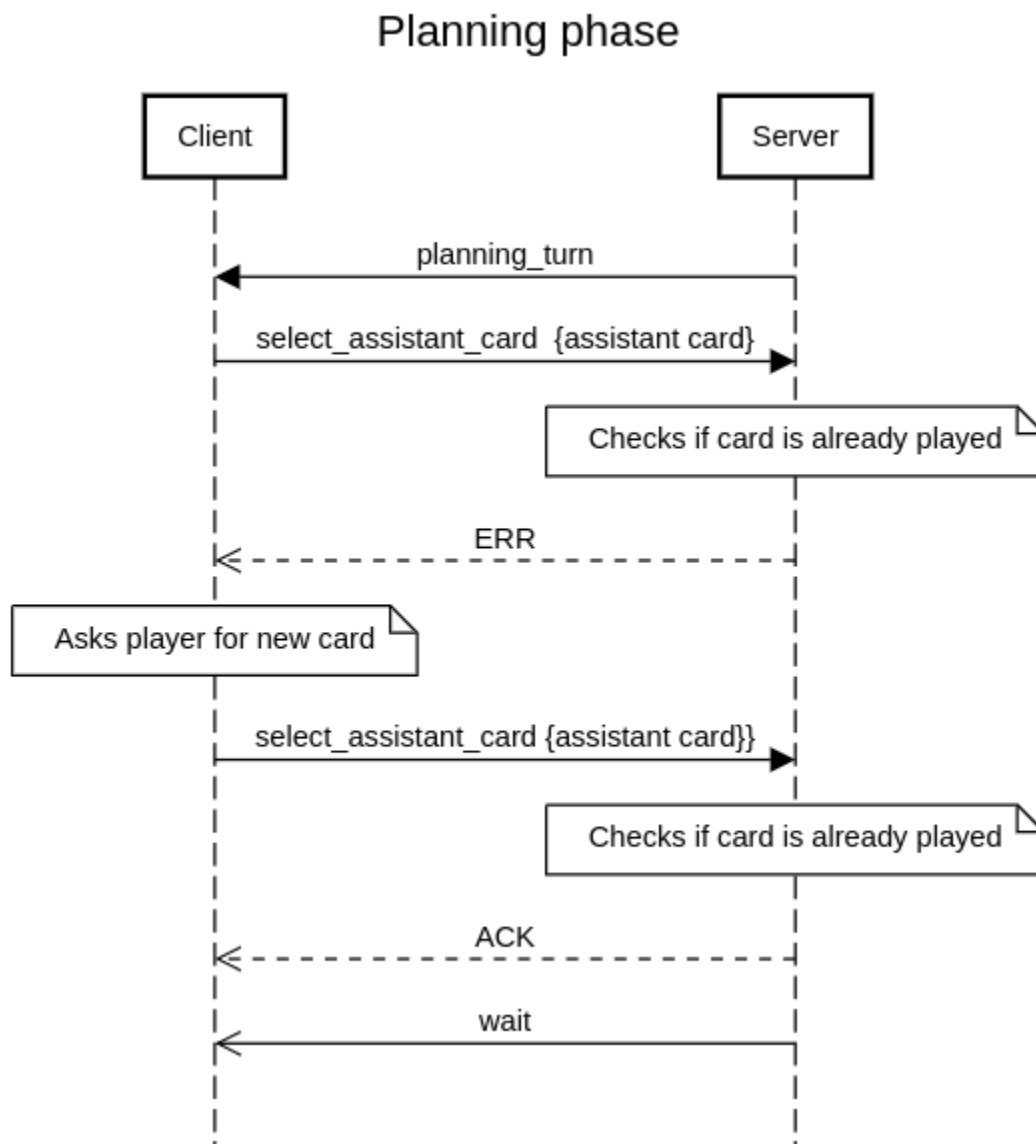
Initial Phase



Planning phase

During the planning phase the server sends messages *planning_turn* to the client which should select their assistant card next and waits for selection. After selection is done, the server checks if there is already the same card played and notifies the player.

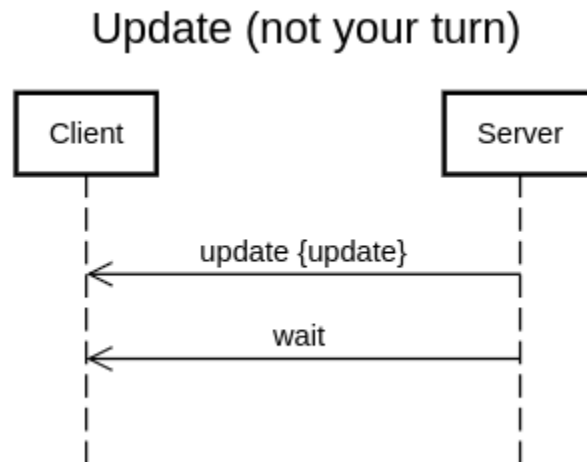
- *planning_turn* - unblocks client and tells it that it is in planning phase
- *select_assistnent_card* {assistant card}
 - *select_assistant_card* - unblocks client and asks it to select assistant card, server expects selection and checks if same card has been played
 - {assistant card} - selected assistant card that player wants to play



Update (not your turn)

This “phase” serves to visualize turns of other players (commands sent to all players except to one who plays next)

- **update** - Notifies client that server has prepared update of information and will send it
- **{update}** - JSON that contains information about islands and player boards (things that are visible to all players)

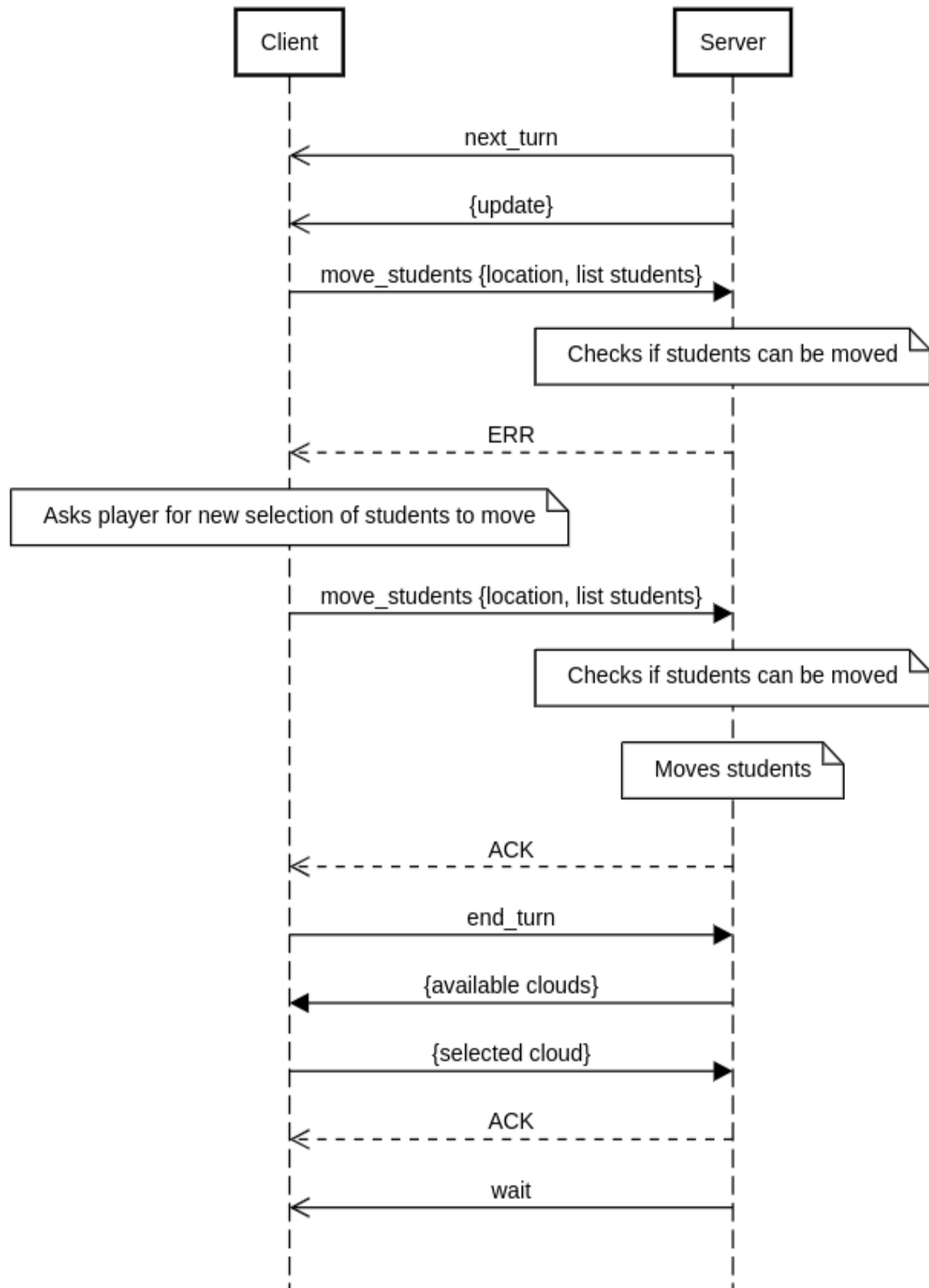


Basics Rules (your turn)

This phase allows player who plays next to move students and select clouds

- **next_turn** - Unlocks client and notifies it that update will be sent and that it is its turn to play
- **move_students {location, list students}**
 - **move_students** - notifies client that server wants to move students, server expects selection and will check if all data is valid. Client expects ACK if not received that means that connection is lost (uses timeout)
 - **{location, list students}** - JSON that contains location to which student should be moved and list of students to be moved (server will check data and send ERR if not valid, otherwise students will be moved)
- **end_turn** - indicates that the client has finished all possible actions : moving students and activating personalities (in advanced game). Server will send list of clouds
- **{available cloud}** - list of available clouds
- **{selected cloud}** - sends selected cloud to server

Basic Rules (your turn)

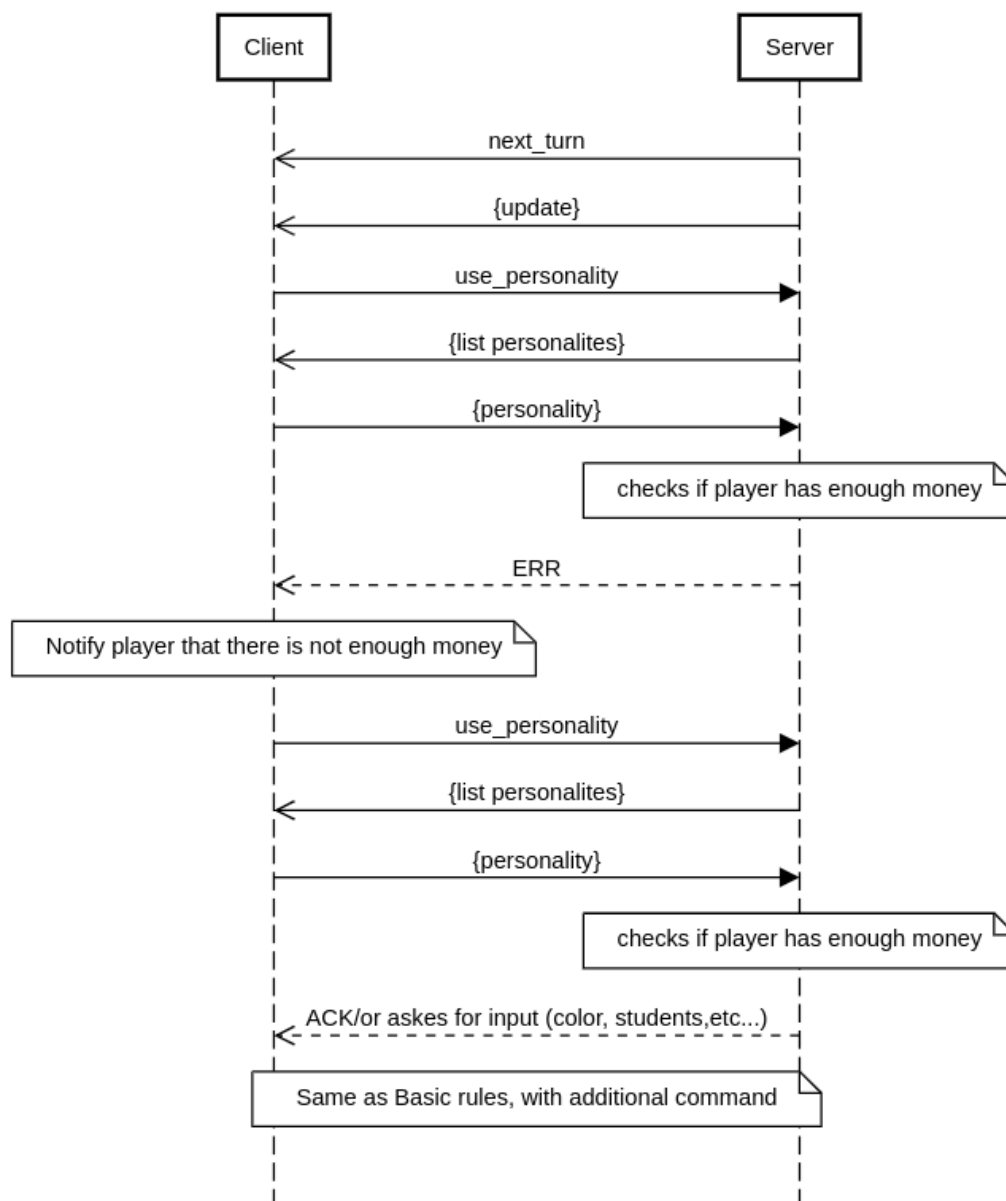


Advanced Rules (your turn)

This phase shows additional "command" use_personality which can be sent by client in moment during its turn

- **use_personality** - Notifies Server that clients wants to use a personality, server prepares list of available personalities
- **{list personalities}** - list of available personalities from which client should choose one
- **{personality}** - selected personality by client, Server will check if player has enough coins
- **ACK/or asks for input(color, students, etc...)** - ACK if there is no additional actions needed, otherwise asks for input and checks its validity

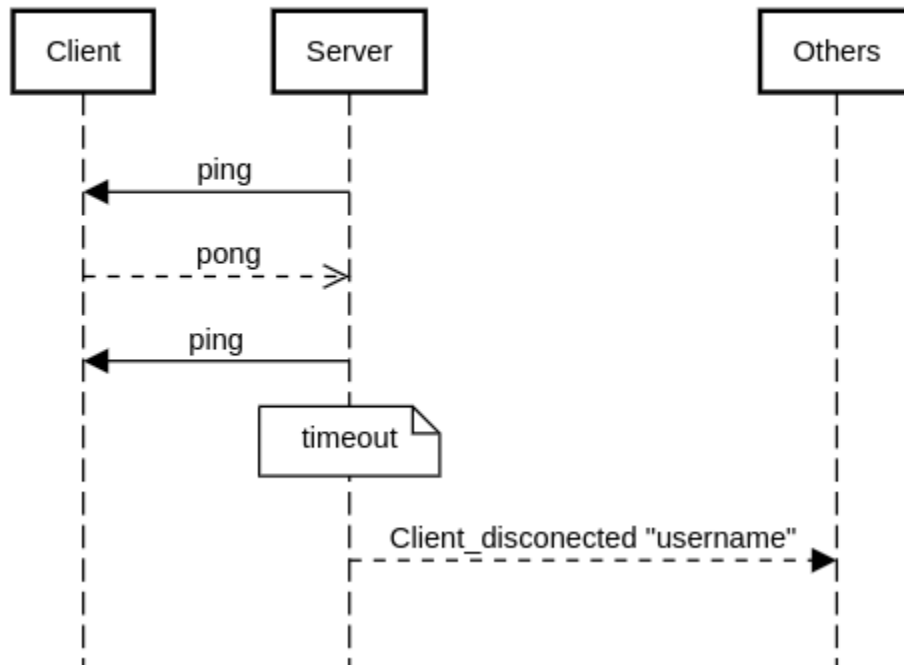
Advanced Rules (your turn)



Connections Check

There is an asynchronous method (in both client and server) which will exclusively send/receive *ping* or *pong*. In case that response is not received server will notify other players that there was disconnection (using *client_disconnected* followed by username of disconnected player)

Connections Check



Four player game

*In case of the game with four players only one player per team should select the color of towers. After deciding who is captain of each team (done by players or randomly by server), the server will send **JSON** containing the id of each team captain.*

