

# Communication Protocol

## Overview

Server and Client communicate through messages. Messages are instances of Message class serialized and then sent via network to be deserialized and read.

Serialization is obtained through java.io.Serializable interface.

All the computing happens in the Server, Clients see the state of the Game but all the modification to the model are done by the server.

## Controller

The Controller is situated in the Server and comprises ClientHandler.java and all the Classes in Controller sub-package. Messages received by the server are parsed by the ClientHandler.java and then processed by the suitable Controller class, which then modify the model and give a response later to be sent to the clients.

## Connection phases

### Connection Phase

Server waits for a Client to connect, when it happens a new ClientHandler thread starts and the connection between client and server is managed by the new thread.

### Login Phase

The only accepted message is a LOGIN message that sets the client username. If any other type of message is received the response is an error message asking to log in.

After login when the username is set the user is allowed to start a game.

### Game Phase

The game phase starts when a player sends the command START, if there are enough available players a new Game is created and the game starts.

### Planning Phase

First half of the game phase, at the beginning of every round each player has to play an Assistant, the only accepted message in this phase is a PLAY message, the response depends on the turns and the model.

### Action Phase

Second half of the game phase, in their turn each player has to complete a set of actions, the accepted commands are PLACE, MOVE, CHOOSE in this order, while USE command is accepted at any moment during the action phase.

## Message structure

Messages are defined by the Message.java class. Every message has 8 arguments:

### Command

Contains a Command Enum that defines the message goal.

argNum1

Contains an Integer. Is used in most of commands and is the main Integer argument.

argNum2

A List of Integer. Contains all the secondary integer arguments.

argString

A String used in LOGIN and STRING commands.

to\_tile

A ToTile Enum used in the PLACE command.

argColour

A Colour Enum, used for some Character effects.

status

If the message is a STATUS message contains a GamePack describing the current state of the game, if not it's null.

standard

Boolean attribute, if the message has a correct structure standard is true, if not it's false and the message doesn't need to be processed since the structure doesn't resemble any existing command.

## Connection anomalies

### Client disconnects

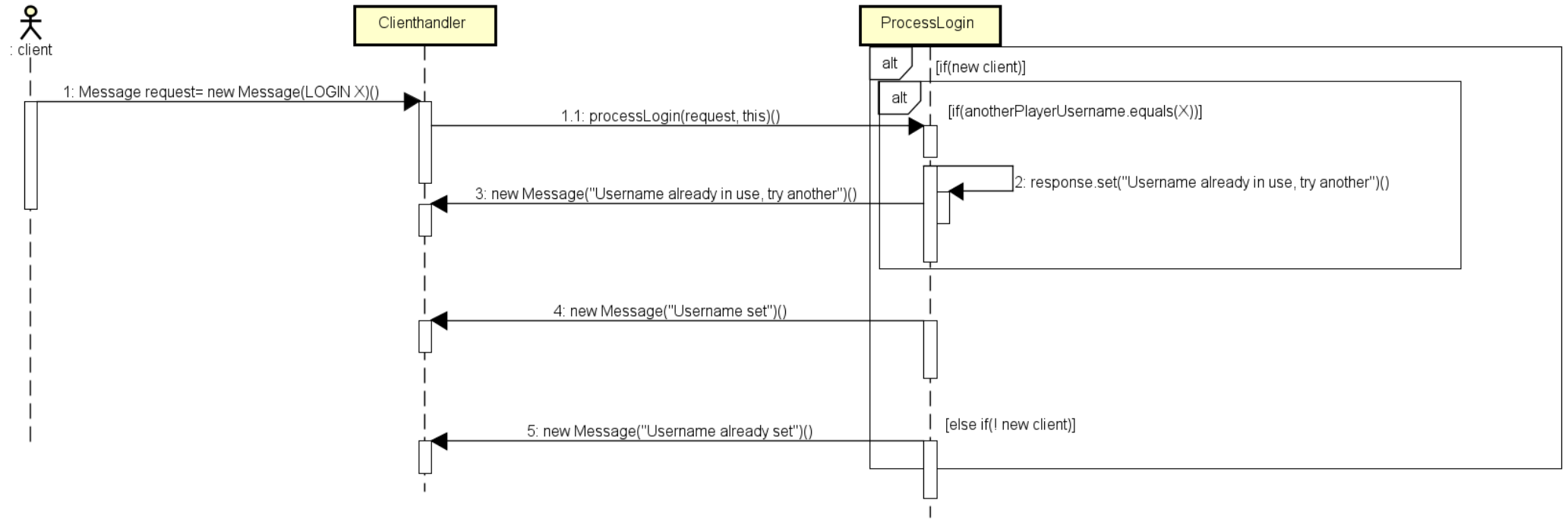
When a client disconnects, either with LOGOUT or because the connection no longer exists, the server processes the logout.

If the client was in an ongoing game the game is disrupted and all the other players are alerted.

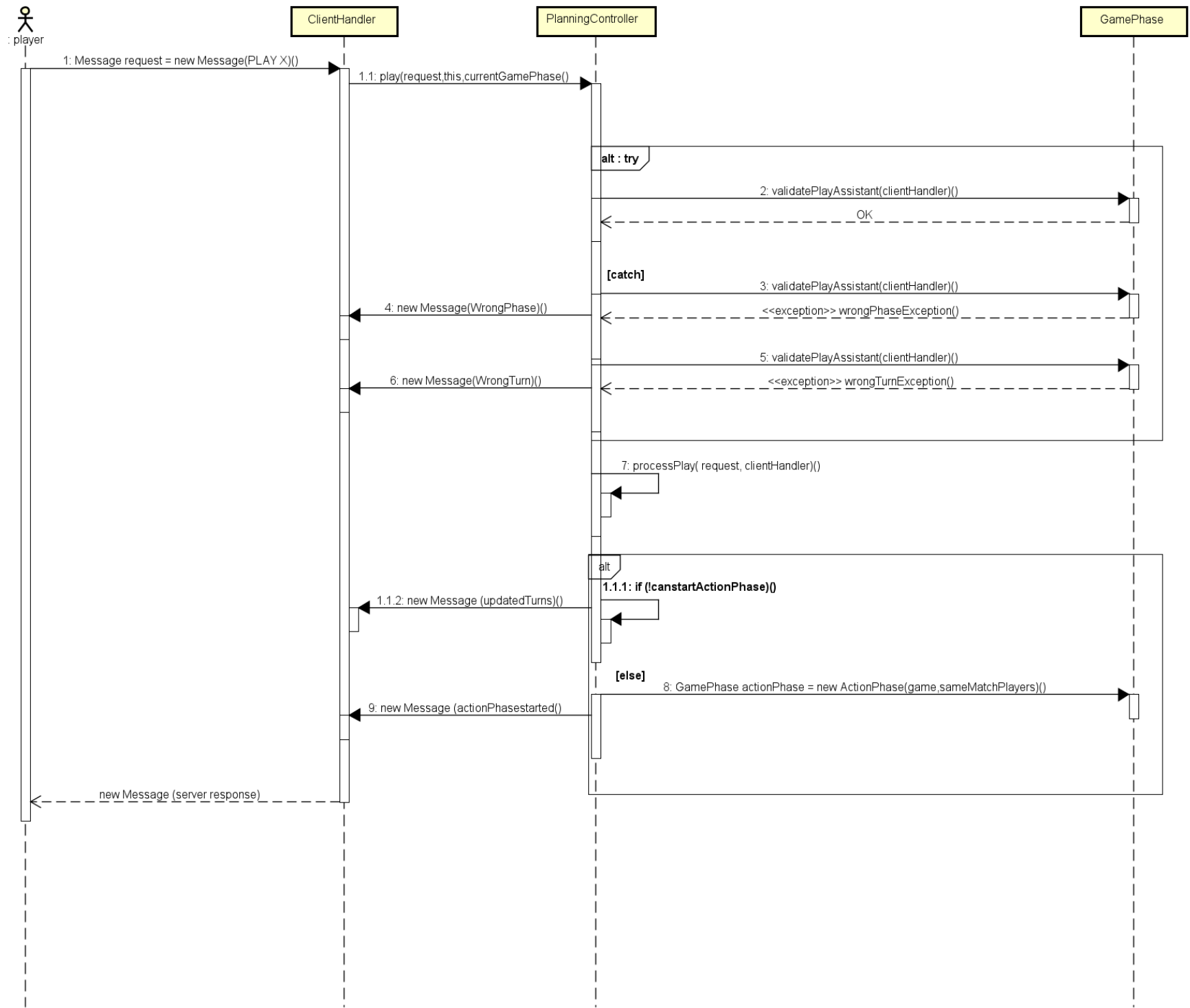
### Server disconnects

If the server disconnects, connected clients try to reconnect. Any ongoing game is disrupted.

sd Sequence Login Action



sd Sequence Diagram of Play action



sd Sequence diagram of Action Phase

