



Scrabble Communication Protocol

This document describes the communication details between the server and the clients in the Scrabble Contest in CMPN402 Spring 2019. The document is divided into 5 sections:

- Websockets.
- Data Types, Formats and Enumerations.
- Scenarios.
- Server State Machine.
- Client State Machine.

Websockets

The communication will be done using Websockets protocol version 13 as defined by RFC6455 (<https://tools.ietf.org/html/rfc6455>). All of the data will be passed using binary messages. The server will be listening over a port that will be specified at the beginning of the contest.

The server will implement a heartbeat system to check for client disconnections. Initially, every client is assumed to be alive and every ping interval, the server will do the following:

- Check if any client is not flagged as alive and terminate its connection.
- Send a ping message to every alive client then remove the alive flag.
- If the client replies with a pong message, it is flagged as alive again.

The ping interval is configurable.

Data Types, Formats and Enumerations

Data types:

- Unsigned 8-bit integer: uint8.
- Signed 32-bit integer (Big Endian): int32 BE.

Tile Formats:

- **Tile-Format-1** is used to denote tiles that are not played yet.
 - 0: no tile
 - [1-26]: tiles [A-Z]
 - 100: blank tile
- **Tile-Format-2** is used to denote tile that are on board.
 - 0: no tile
 - [1-26]: tile [A-Z]
 - [101-126]: blank tiles played as tiles from [A-Z].

Table 1: Enumerations

Enumerations		
Message Types	Server States	Client States
NAME (0)	INIT	INIT
START (1)	READY	READY
PASS (2)	IDLE	IDLE
EXCHANGE (3)	AWAIT_CHALLENGE	THINKING
PLAY (4)		AWAIT_EXCHANGE_REPOSNSE
NO_CHALLENGE (5)		AWAIT_PLAY_RESPONSE
CHALLENGE (6)		AWAIT_AGENT_CHALLENGE
CHALLENGE_ACCEPTED (7)		AWAIT_CHALLENGE_REPONSE
CHALLENGE_REJECTED (8)		
INVALID (9)		
END (10)		

Scenarios

Initialization: Any new client will receive any empty NAME message to which it should reply with a NAME message containing its team name. After two clients have done this process, the server and both clients should be ready to start a game.

Starting a game: The server will send a START message containing the board state, 7 tiles, players' order and time information to both player. The player with order 1 should start playing.

Ending a game: At any moment, the client should expect an END message which means that the game has ended. The message will include the final scores and the reason for ending the game (No more tiles, time ended, connection error or stop button pressed).

Passing a turn: if a player wants to pass their turn, they should send an empty PASS message to the server and the server will send a PASS message to the other player containing their remaining time and the total remaining time. The two player will then be swapped.

Exchanging tiles: if a player wants to exchange some or all of their tiles, they should send an EXCHANGE message containing the tiles to be exchanged to the server. If the move is invalid, the server will reply with an INVALID message containing their remaining time and the total remaining time. Otherwise, the server will send an EXCHANGE message containing new tiles. The server will also send the other player an EXCHANGE message containing their remaining time and the total remaining time. If the move is valid, the two player will then be swapped.

Play tiles: if a player wants to place tiles on the board, they should send a PLAY message containing the row, column and direction in addition to the tiles and gained points. If the move is invalid (the server will not check the validity of the newly created words), the server will reply with an INVALID message containing their remaining time and the total remaining time. Otherwise, the server will echo the PLAY message to the other player in addition to the time allowed for a challenge, their remaining time and the total remaining time. If the move is valid, the two player will then be swapped and the server will wait for the current player's reply which can be one of the following (The server will pause the game timer and start the challenge timer):

- NO_CHALLENGE: this message is empty and the server will echo it to the other player including their new tiles then it will stop the challenge timer and unpauses the game timer.
- CHALLENGE: this message is also empty and the server will then check the validity of the newly created words (In both cases, the server will stop the challenge timer and unpauses the game timer):
 - If any of the words are invalid, it will send a CHALLENGE_ACCEPTED to both players. For the other player, it will be empty and for the current player, it will contain their remaining time and the total remaining time.

- If all the words are valid, it will send a CHALLENGE_REJECTED to both players. For the current player, it will be empty and for the other player, it will contain their new tiles, their remaining time and the total remaining time. The two player will then be swapped.
- If the challenge time is out before the server receives a reply, the server will default to NO_CHALLENGE and will send NO_CHALLENGE to both clients. For the current player, it will contain their remaining time and the total remaining time and for the other player, it will contain their new tiles. Then the server will stop the challenge timer and unpause the game timer.

Disconnection: Any action except PLAY will cause the server to save a new checkpoint. If one or more client disconnects from the server, the server will save the remaining time for both players into the last checkpoint, and send END to connected clients with Reason=2 (Connection Error) then it will go to the initial state to listen for connection requests. Clients that receive the END message should stay ready for the game to continue. The disconnected client should request a new connection then go through the initialization sequence. After both clients are connected, the server can resume the game (if pause button is pressed) and send a new START message containing the checkpoint state to both clients.

Server State Machine

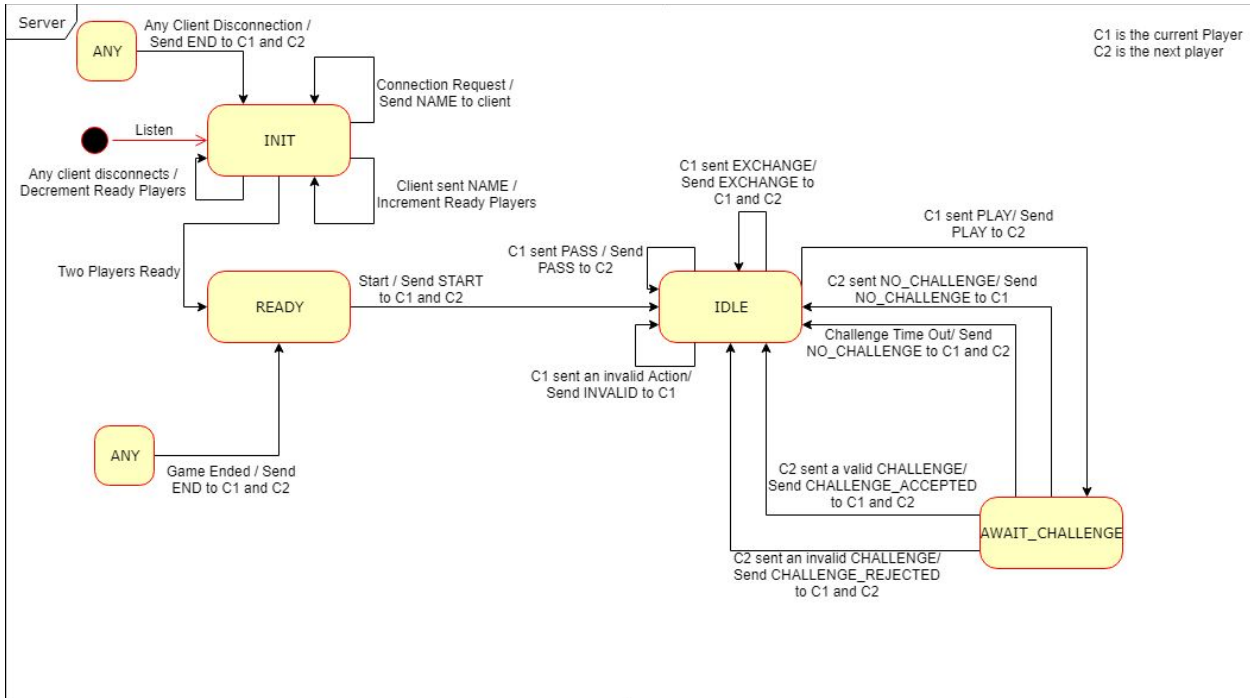


Figure 1: Server State Machine

State: INIT

- Input: Connection request from Client. Goto: **INIT**
 - Assign Client to an empty player slot
 - Send NAME to client
 - Type (uint8): message type. Always=0.
- Input: Received NAME from Client. Goto: **INIT**
 - Expected Format:
 - Type (uint8): message type. Always=0.
 - Name (uint8 xN): name is utf-8 format where N is the name length.
 - Increment number of ready players.
- Input: Any client disconnects. Goto: **INIT**
 - Decrement number of ready players.
- Input: Number of ready players became 2. Goto: **READY**.

State: READY

- Input: Start Game Button Clicked or Recovery from Connection Error. Goto: **IDLE**
 - Send START to C1 and C2
 - Type (uint8): message type. Always=1.
 - Order (uint8): 1 means you are 1st player, 2 means you are 2nd player.

- Tiles (uint8 x7): Starting tiles (Tile-Format-1).
 - Board (uint8 x15x15): Board state in row-major (Tile-Format-2).
 - Score (int32 BE): your final score.
 - Opponent Score (int32 BE): your opponent's final score.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Start Game Timer

State: IDLE

- Input: Received PASS from C1. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=2.
 - Send PASS to C2
 - Type (uint8): message type. Always=2.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
 - Swap C1 and C2.
- Input: Received EXCHANGE or PLAY from C1 and it is invalid. Goto: **IDLE**
 - Expected Format: Discussed in the next two inputs.
 - Send INVALID to C1
 - Type (uint8): message type. Always=9.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received EXCHANGE from C1 and it is valid. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=3.
 - Tiles (uint8 x7): tiles to be exchanged (Tile-Format-1).
 - Send EXCHANGE to C1
 - Type (uint8): message type. Always=3.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
 - Send EXCHANGE to C2
 - Type (uint8): message type. Always=3.
 - Count (uint8): number of tiles exchanged.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
 - Swap C1 and C2.
- Input: Received PLAY from C1 and it is valid. Goto: **AWAIT_CHALLENGE**
 - Expected Format:
 - Type (uint8): message type. Always=4.
 - Column (uint8): starting column. Valid values are [1-15].
 - Row (uint8): starting row. Valid values are [1-15].
 - Direction (uint8): direction of play. 0 means Right and 1 means Down.

- Tiles (uint8 x7): tiles to be played (Tile-Format-2).
 - Score (int32 BE): collected points.
- Send PLAY to C2
 - Same as input.
 - Challenge Time (int32 BE): allowed challenge time in milliseconds.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Swap C1 and C2, pause Game timer and start Challenge timer.

State: **AWAIT_CHALLENGE**

- Input: Received NO_CHALLENGE from C1. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=5.
 - Send NO_CHALLENGE to C2.
 - Type (uint8): message type. Always=5.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
 - Stop Challenge timer and unpause Game timer.
- Input: Received CHALLENGE from C1 and it is valid. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=6.
 - Send CHALLENGE_ACCEPTED to C2.
 - Type (uint8): message type. Always=7.
 - Send CHALLENGE_ACCEPTED to C1.
 - Type (uint8): message type. Always=7.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
 - Stop Challenge timer and unpause Game timer.
- Input: Received CHALLENGE from C1 and it is invalid. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=6.
 - Send CHALLENGE_REJECTED to C2.
 - Type (uint8): message type. Always=8.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
 - Send CHALLENGE_REJECTED to C1.
 - Type (uint8): message type. Always=8.
 - Swap C1 and C2, stop Challenge timer and unpause Game timer..
- Input: Challenge Time Out. Goto: **IDLE**
 - Send NO_CHALLENGE to C1
 - Type (uint8): message type. Always=5.
 - Player Time (int32 BE): your remaining time in milliseconds.

- Total Time (int32 BE): total remaining time in milliseconds.
- Send NO_CHALLENGE to C2
 - Type (uint8): message type. Always=5.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
- Stop Challenge timer and unpause Game timer.

State: * **ANY**

- Input: Game ending condition occurred. Goto: **READY**
 - Send END to C1 and C2.
 - Type (uint8): message type. Always=10.
 - Reason (uint8): gamed-end reason. 0=All tiles used, 1=Time ended, 2=Connection Error, 3=Stop button pressed.
 - Score (int32 BE): your final score.
 - Opponent Score (int32 BE): your opponent's final score.
- Input: Disconnection from any of the clients. Goto: **INIT**
 - Send END to any connected client with Reason=2.
 - Same as previous input.

Client State Machine

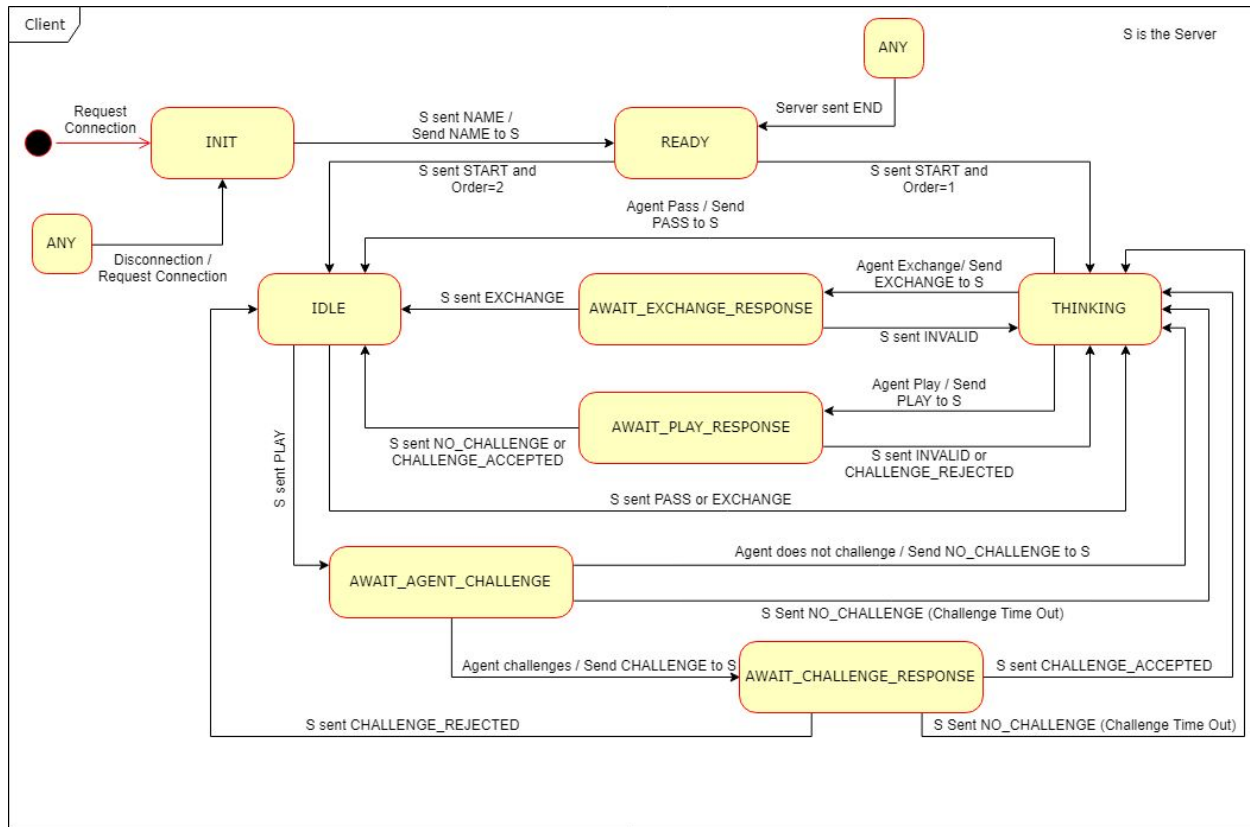


Figure 2: Client State Machine

State: INIT

- Input: Received NAME from S. Goto: **READY**
 - Expected Format:
 - Type (uint8): message type. Always=0.
 - Send NAME to S.
 - Type (uint8): message type. Always=0.
 - Name (uint8 xN): name is utf-8 format where N is the name length.

State: READY

- Input: Received START from S and Order=1. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=1.
 - Order (uint8): 1 means you are 1st player, 2 means you are 2nd player.
 - Tiles (uint8 x7): Starting tiles (Tile-Format-1).
 - Board (uint8 x15x15): Board state in row-major (Tile-Format-2).
 - Score (int32 BE): your final score.

- Opponent Score (int32 BE): your opponent's final score.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received START from S and Order=2. Goto: **IDLE**
 - Expected Format: Same as previous input.

State: THINKING

- Input: Agent plays PASS. Goto: **IDLE**
 - Send PASS to S.
 - Type (uint8): message type. Always=2.
- Input: Agent plays EXCHANGE. Goto: **AWAIT_EXCHANGE_RESPONSE**
 - Send EXCHANGE to S.
 - Type (uint8): message type. Always=3.
 - Tiles (uint8 x7): tiles to be exchanged (Tile-Format-1).
- Input: Agent plays PLAY. Goto: **AWAIT_PLAY_RESPONSE**
 - Send PLAY to S.
 - Type (uint8): message type. Always=4.
 - Column (uint8): starting column. Valid values are [1-15].
 - Row (uint8): starting row. Valid values are [1-15].
 - Direction (uint8): direction of play. 0 means Right and 1 means Down.
 - Tiles (uint8 x7): tiles to be played (Tile-Format-2).
 - Score (int32 BE): collected points.

State: AWAIT_EXCHANGE_RESPONSE

- Input: Received INVALID from S. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=9.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received EXCHANGE from S. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=3.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).

State: AWAIT_PLAY_RESPONSE

- Input: Received INVALID from S. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=9.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received CHALLENGE_REJECTED from S. Goto: **THINKING**
 - Expected Format:

- Type (uint8): message type. Always=8.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received NO_CHALLENGE from S. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=5.
 - Tiles (uint8 x7): new tiles (Tile-Format-1).
- Input: Received CHALLENGE_ACCEPTED from S. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=7.

State: IDLE

- Input: Received PASS from S. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=2.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received EXCHANGE from S. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=3.
 - Count (uint8): number of tiles exchanged.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received PLAY from S. Goto: **AWAIT_AGENT_CHALLENGE**
 - Expected Format:
 - Type (uint8): message type. Always=4.
 - Column (uint8): starting column. Valid values are [1-15].
 - Row (uint8): starting row. Valid values are [1-15].
 - Direction (uint8): direction of play. 0 means Right and 1 means Down.
 - Tiles (uint8 x7): tiles to be played (Tile-Format-2).
 - Score (int32 BE): collected points.
 - Challenge Time (int32 BE): allowed challenge time in milliseconds.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.

State: AWAIT_AGENT_CHALLENGE

- Input: Agent do not want to challenge. Goto: **THINKING**
 - Send NO_CHALLENGE to S.
 - Type (uint8): message type. Always=5.
- Input: Agent want to challenge. Goto: **AWAIT_CHALLENGE_RESPONSE**
 - Send CHALLENGE to S.
 - Type (uint8): message type. Always=6.

- Input: Server sent NO_CHALLENGE. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=5.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.

State: **AWAIT_CHALLENGE_RESPONSE**

- Input: Server sent NO_CHALLENGE. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=5.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received CHALLENGE_ACCEPTED from S. Goto: **THINKING**
 - Expected Format:
 - Type (uint8): message type. Always=7.
 - Player Time (int32 BE): your remaining time in milliseconds.
 - Total Time (int32 BE): total remaining time in milliseconds.
- Input: Received CHALLENGE_REJECTED from S. Goto: **IDLE**
 - Expected Format:
 - Type (uint8): message type. Always=8.

State: *** ANY**

- Input: Received END from S. Goto: **READY**
 - Expected Format:
 - Type (uint8): message type. Always=10.
 - Reason (uint8): gamed-end reason. 0=All tiles used, 1=Time ended, 2=Connection Error, 3=Stop button pressed.
 - Score (int32 BE): your final score.
 - Opponent Score (int32 BE): your opponent's final score.
- Input: Disconnection from Server. Goto: **INIT**
 - Request new connection.