

Scrabble Game Report

COMP90015_2018_SM2: Distributed Systems

Zhengyu Chen(ID:991678 zhengyuc@student.unimelb.edu.au)

Rongxin Zhu(ID:938816, rongxinz1@student.unimelb.edu.au)

Kai Liu(ID:882063, kail6@student.unimelb.edu.au)

Li Shen(ID:1001920, shels@student.unimelb.edu.au)

Tutor: Lakshmi Jagathamma Mohan

1. Introduction

This report talks about the process of designing and implementation of an online multi-player Scrabble game. It implements a multi-threaded server using Java, which allows multiple clients to play together in a same game.

2. System Component

The system implements the distributed Client/Server structure. In general, it consists of three parts: game server, client GUI and client controller. Game server works as an intermediary, which could broadcast message to all users in the game, such as updating the board of game and showing score of all users; or deliver message to single user, such as enabling the operation of current user. Client GUI displays the user interface to every user and listens for user's operations. Client controller handles operations in the part of client GUI from the user and transaction with server.

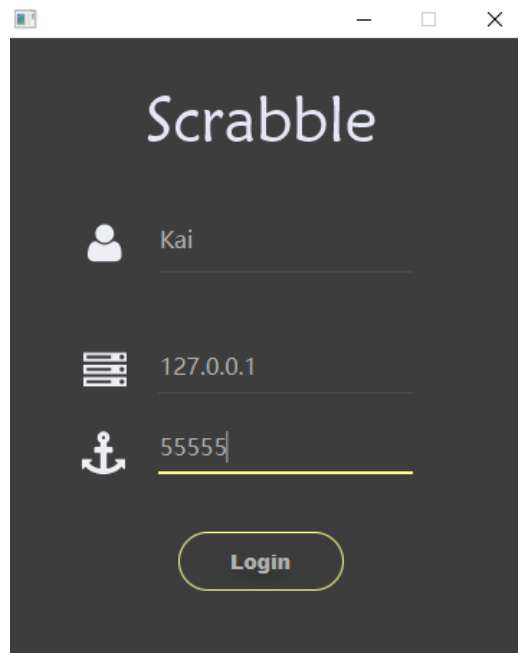
2.1 Communication Protocol

This project uses thread-per-connection architecture for server to communicate with each client. TCP is used, in this project, to make sure every request of every client reaches the server, because quite a lot of operations will be concurrent when multiple clients participate in. If some messages are lost, confusions and errors with regard of words in dictionary under operating may occur.

2.2 Function Implementation

1) Login

Once the server is up, player could enter server IP address, port number, and a unique username according to UI. If either IP or port number is wrong, the UI will pop a warning windows to noticed player re-enter the information, or if the username was already taken, it will ask user change to another name.



2) Waiting Lobby

When a player success login, he will enter game lobby. Game lobby is a place that could see all current online player on the left side bar. The table on right side contains 2 buttons: start and join. Player also can invite another player by click invite button on left of invited player username. The maximum player for a game is four people, so, once the game started, all player on table will start game and display game panel UI.



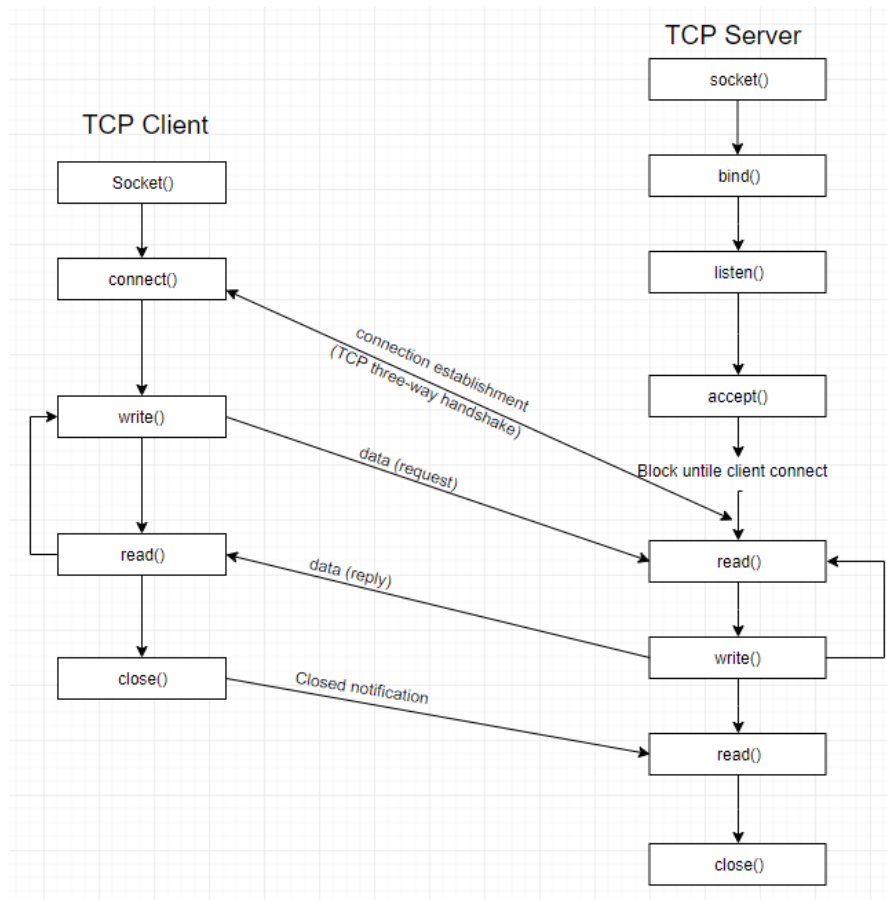
3) Game Panel

After one of table player click start button, everyone will switch to scrabble game UI.

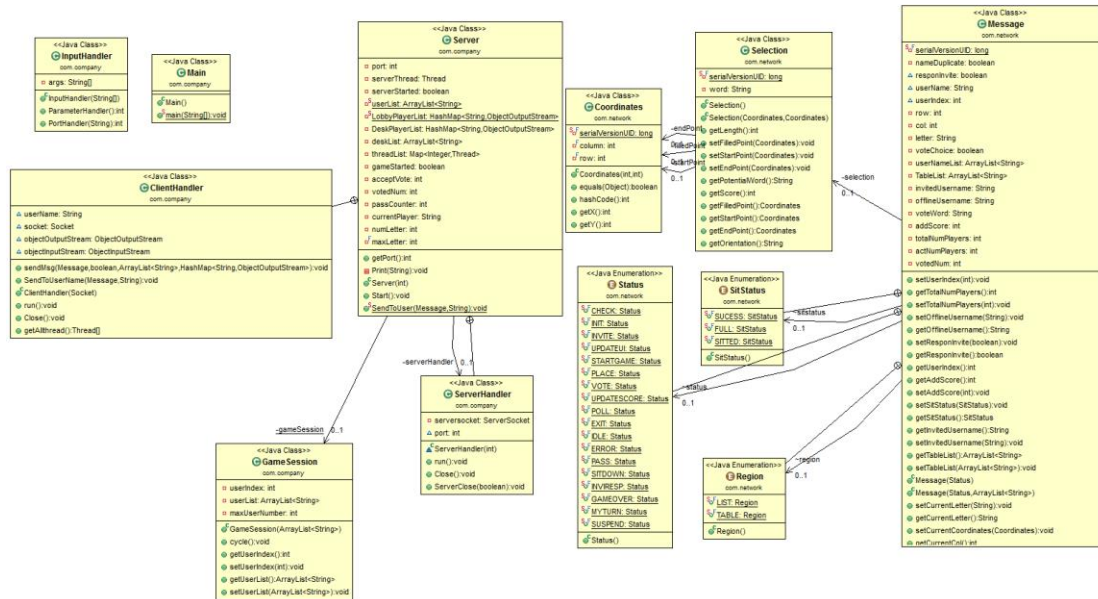
This game is a turn-based game, every round will only accept one player's operation, and all other players could only wait for vote or self-round. When someone's turn, he could either choose to insert a letter or pass this round. If user insert a letter, it will immediately display to all other players. The user can also see two available vote words option for him, once he chooses one, then other users will start vote. Only all other players vote accept, current round player could receive the point. There are three way could end the game: 1. All player pass their turn 2. Full panel was inserted with letter 3. Any player was disconnected.

3. Diagram

3.1 Flow Chart



3.2 Client UML



4. Error Handle

When all players are waiting in the lobby, left side bar could display username list. Now, if there is a user close his tab or disconnected, server will handle the error and broadcasting someone is disconnected and refresh the rest player's UI. Same to game panel, server could detect any player is disconnected and throw the error, we managed the error by let server broadcasting the message to other players and end the game.

If either IP or port number is wrong, the UI will pop a warning windows to noticed player re-enter the information, or if the username was already taken, it will ask user change to another name.

When server crashed, players will be noticed and end the game immediately.

During the game, if it's not the player's turn, he cannot do anything until he is asked to vote confirm or decline, or wait for his turn

5. Contribution

Our group including: Rongxin Zhu, Li Shen, Kai Liu, and Zhengyu Chen. We split our task into two parts, client and server. Kai Liu and Zhengyu Chen are in charge of the server part, and Rongxin Zhu and Li Shen are in charge of the client part. Since the client part has far more work than the server part, Kai Liu and Zhengyu Chen also help write the report and the code of building game session, which is the controller for players' playing in turn.