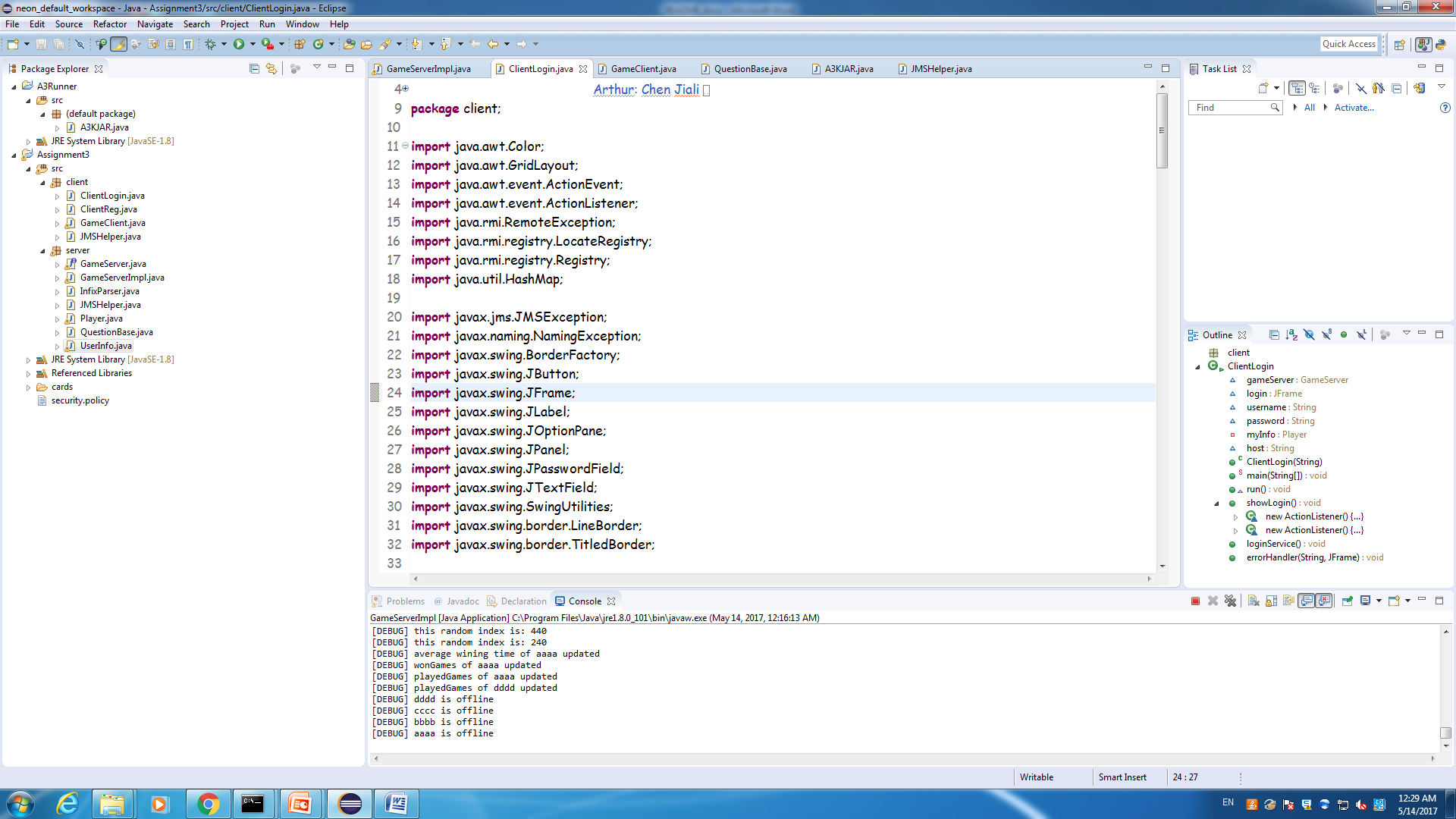
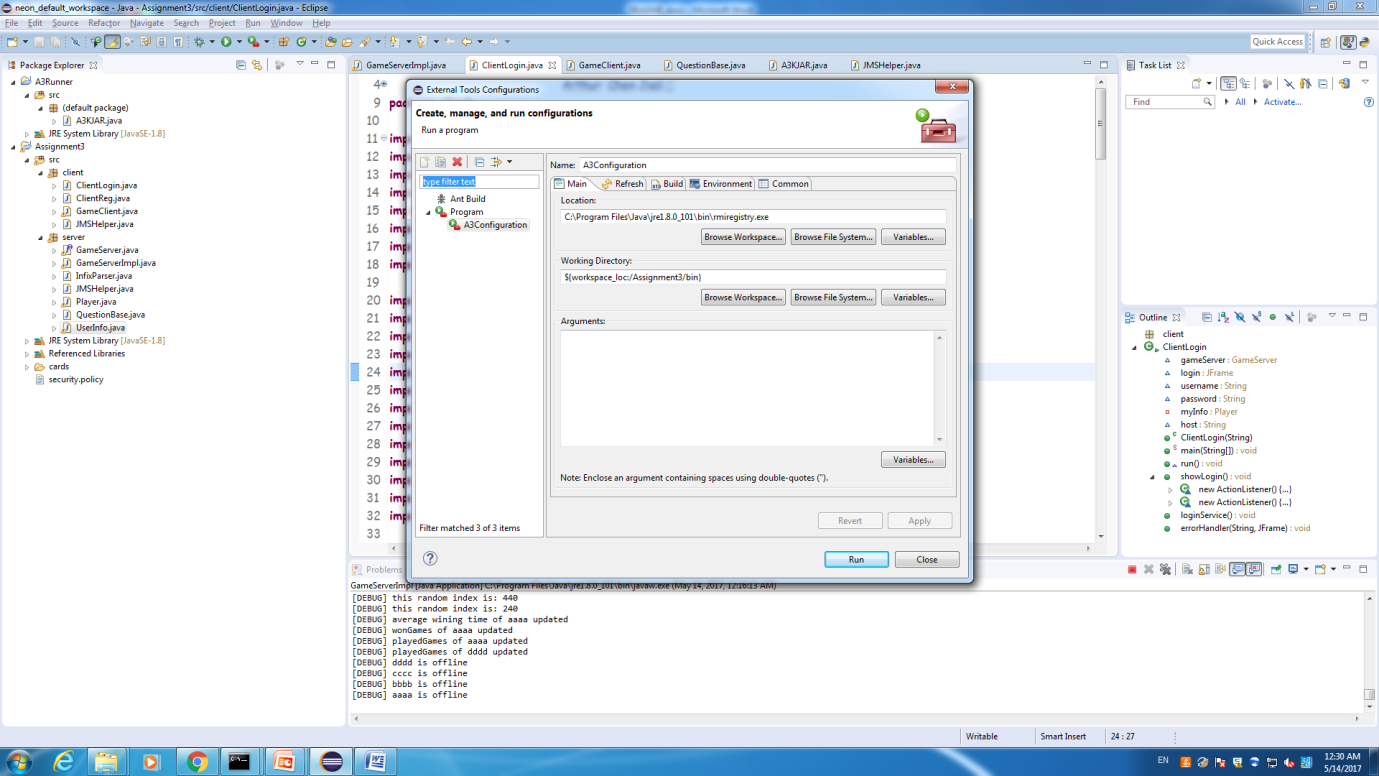
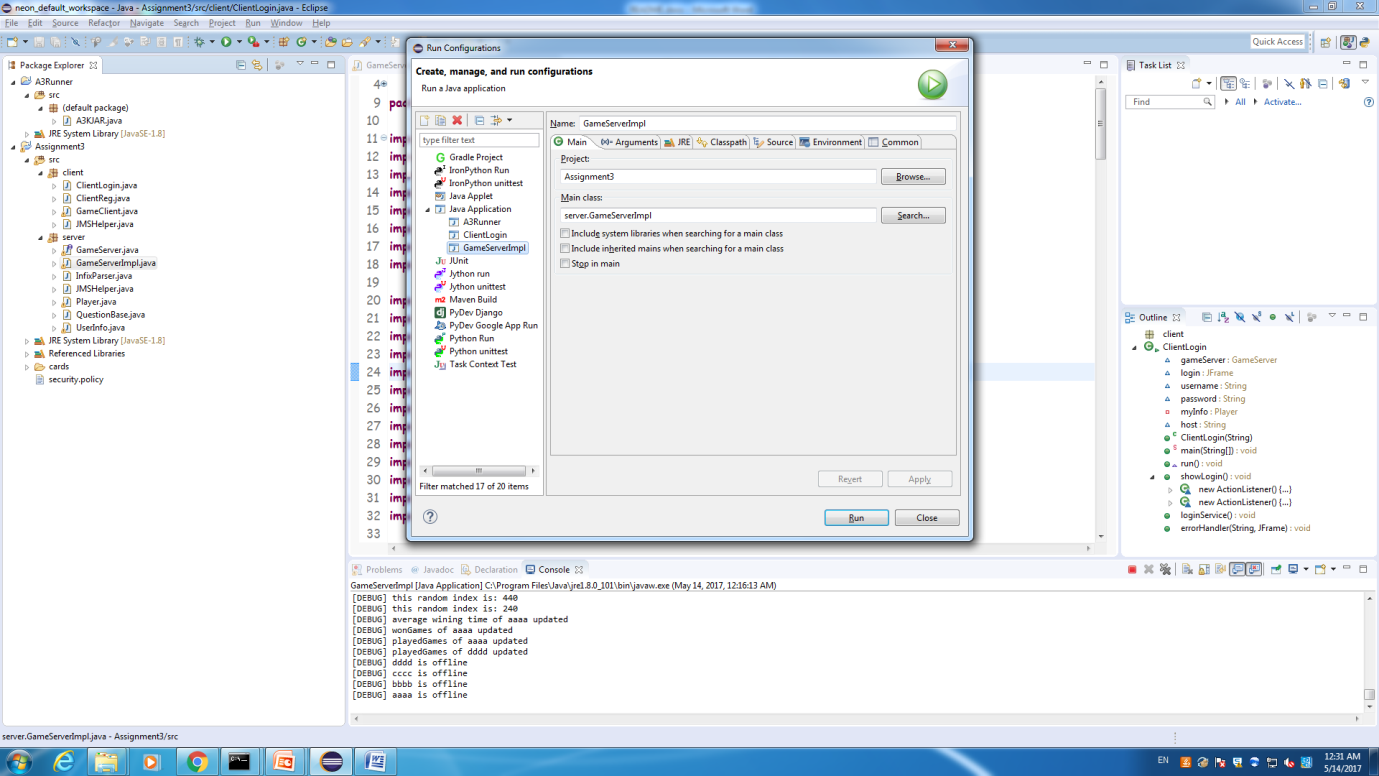
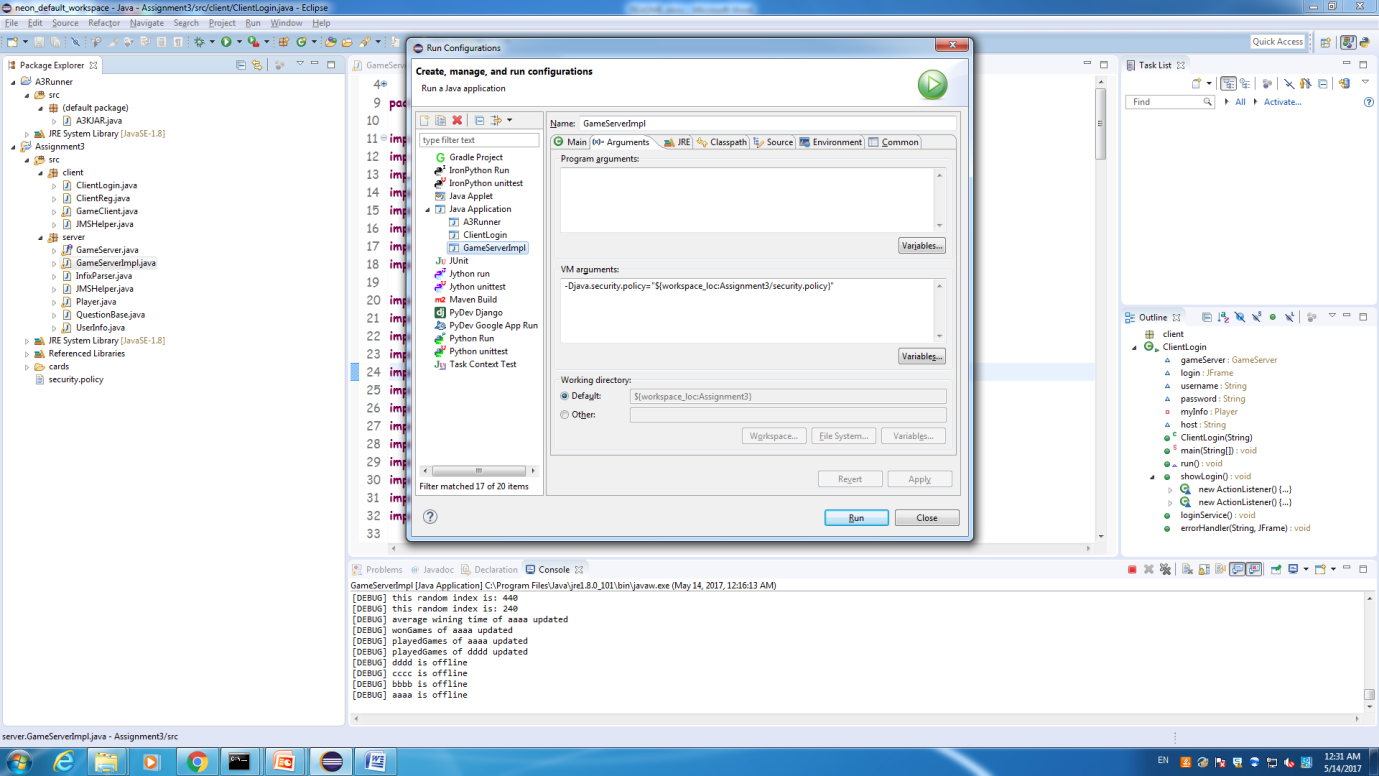
1. Execution

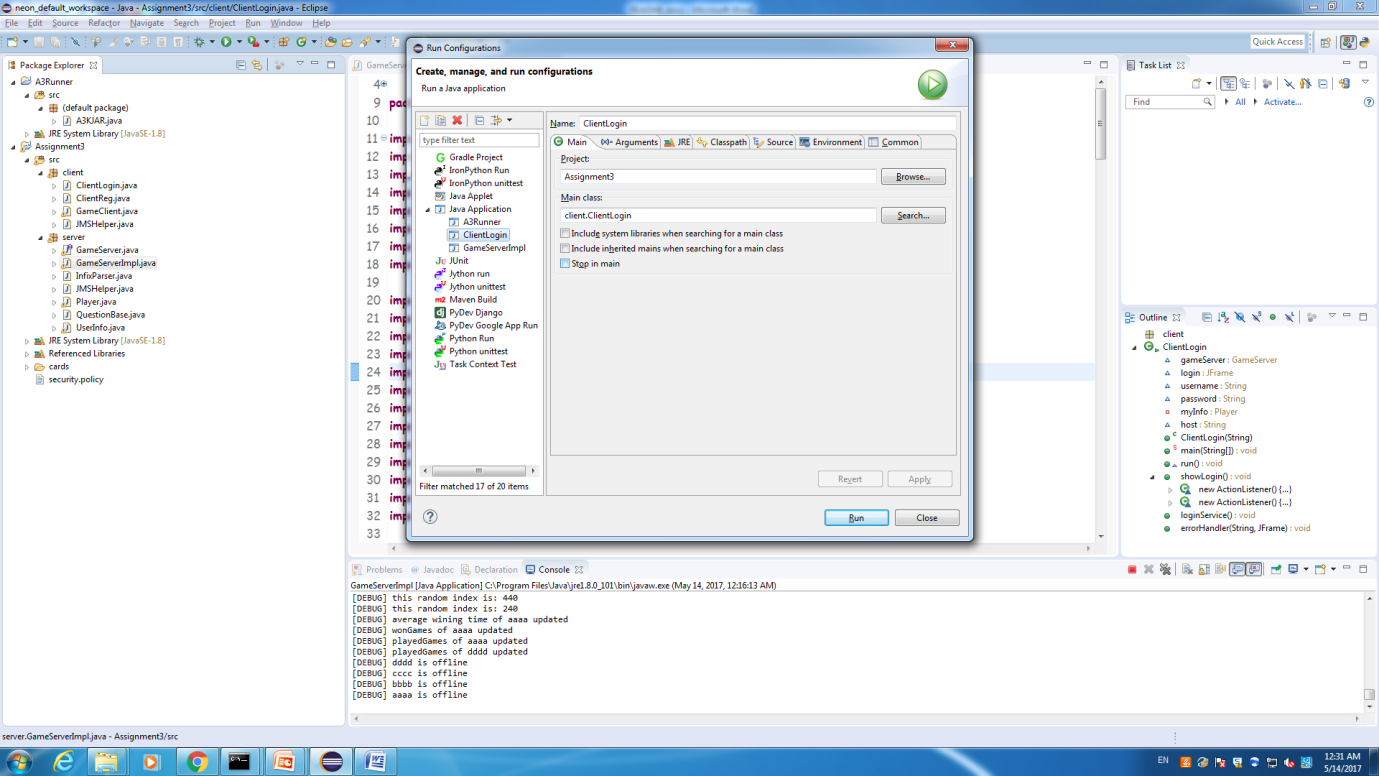


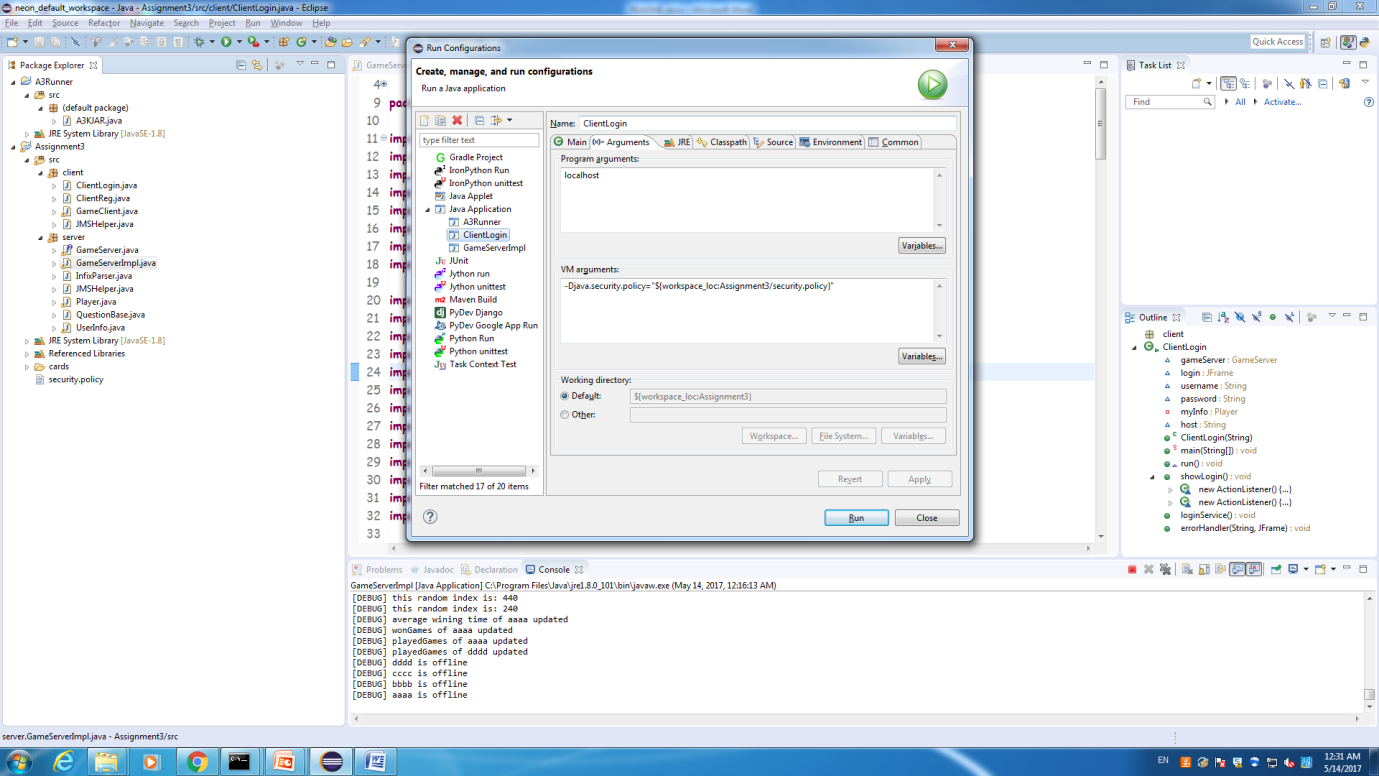
Files should be located like illustrated above.











Here are the configurations for execution.

Afterwards, we need to start the server, and start >2 clients for playing.

1. Code Structure
2. Server Side

**GameServer.java:** this is the interface for the server side; provide interfaces for RMI from the client side.

**GameServerImpl.java:** this is the implementation of the server side, major logic for server. For more details, please refer to the comment and java doc inside.

**InfixParser.java:** this is a functional class providing evaluation of an infix expression

**QuestionBase.java:** this is a functional class providing question base for the server.

**Player.java:** this is a model class for the player class.

**UserInfo.java:** this is a model class for all the users stored. The UserInfo class will be charge of communication with the DB for user information and online user management.

1. Client Side

**ClientLogin.java:** class deal with issues for client login

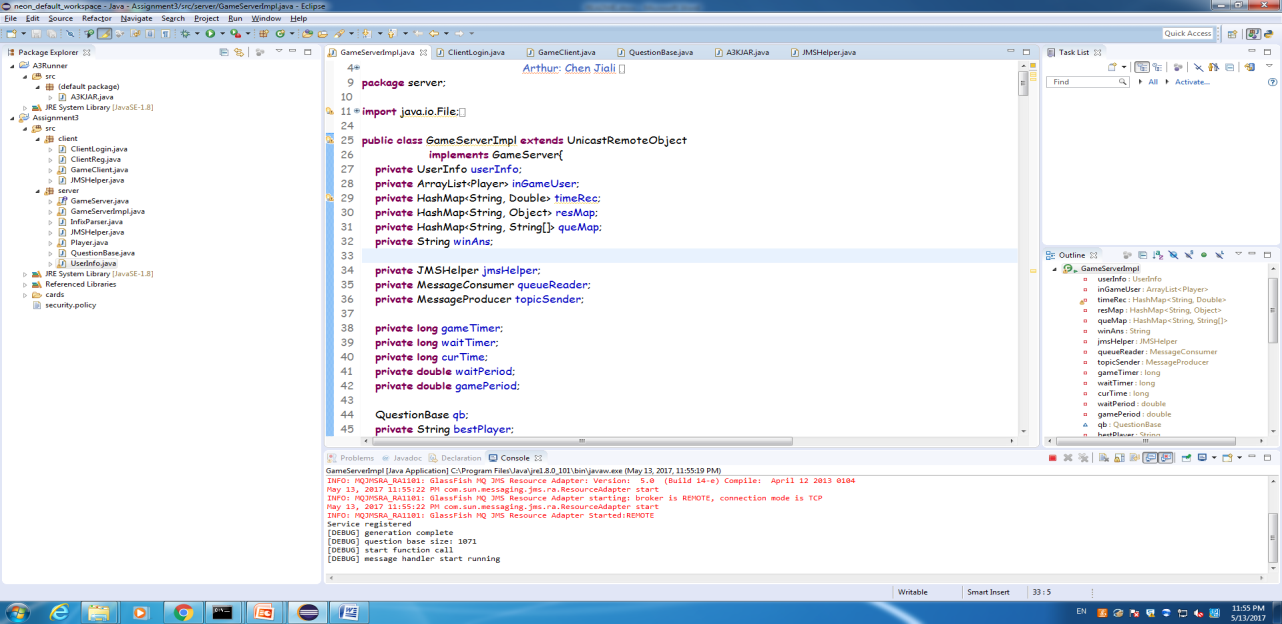
**ClientReg.java:** class deal with issues for client registry

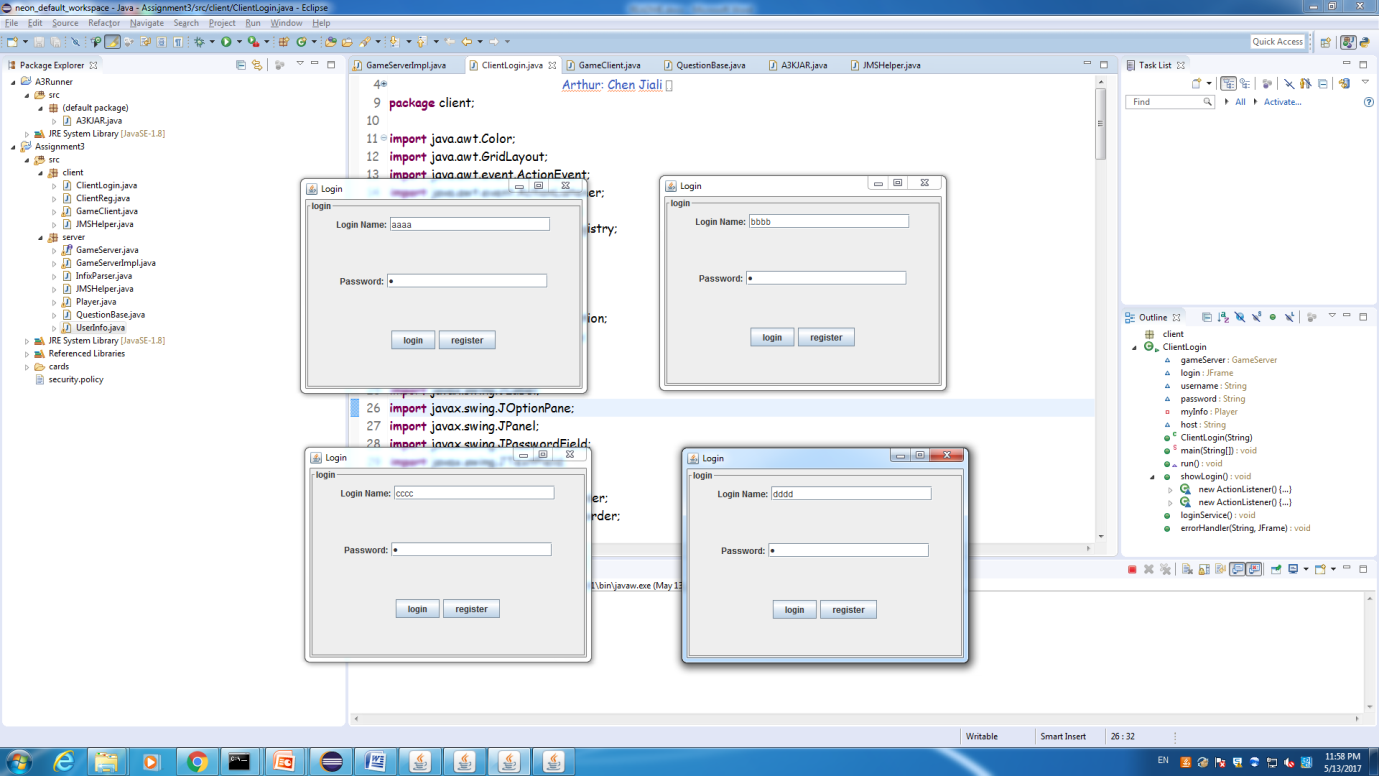
**GameClient.java:** major logic for the client, for details, please kindly refer to the comments and java doc in the file

1. Execution Screen-shoot

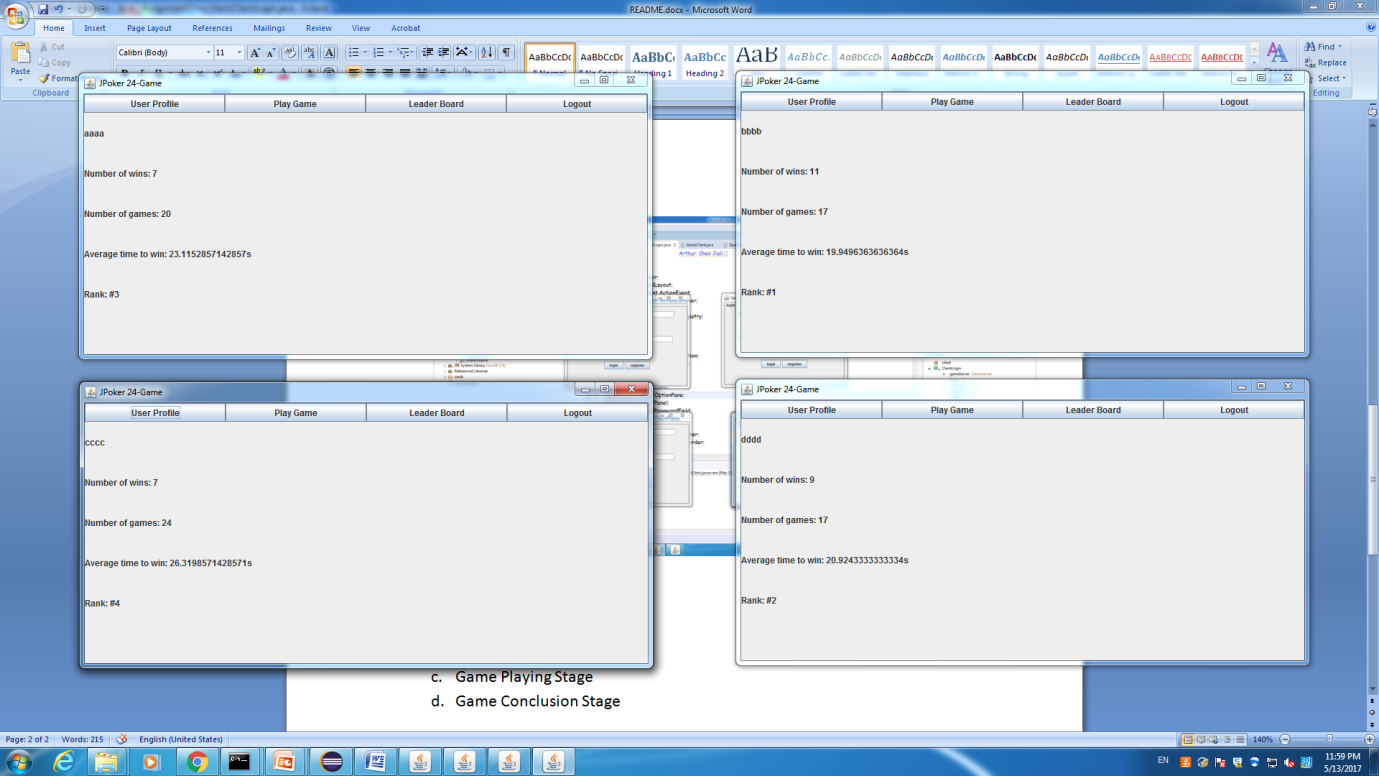
*(Screen-shoots for client login and register are tasks for assignment1, therefore, they are omitted in the readme file for assignment3, the logic for login and registry is identical to one in assignment1 except now the communication is all with the database)*

1. Initial Stage

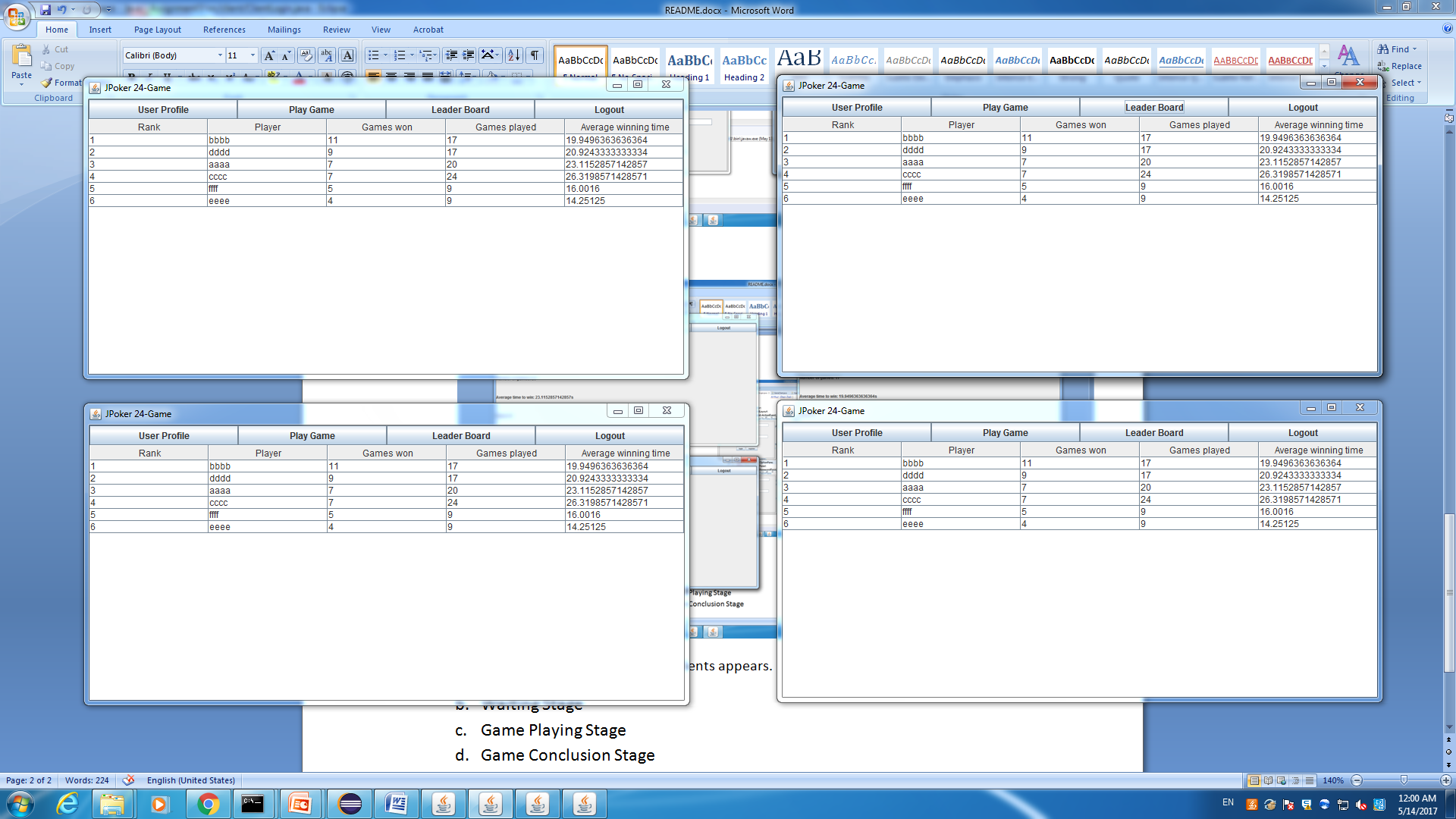
The server starts.



Login with four clients.



After login, the profile for the four clients appears.

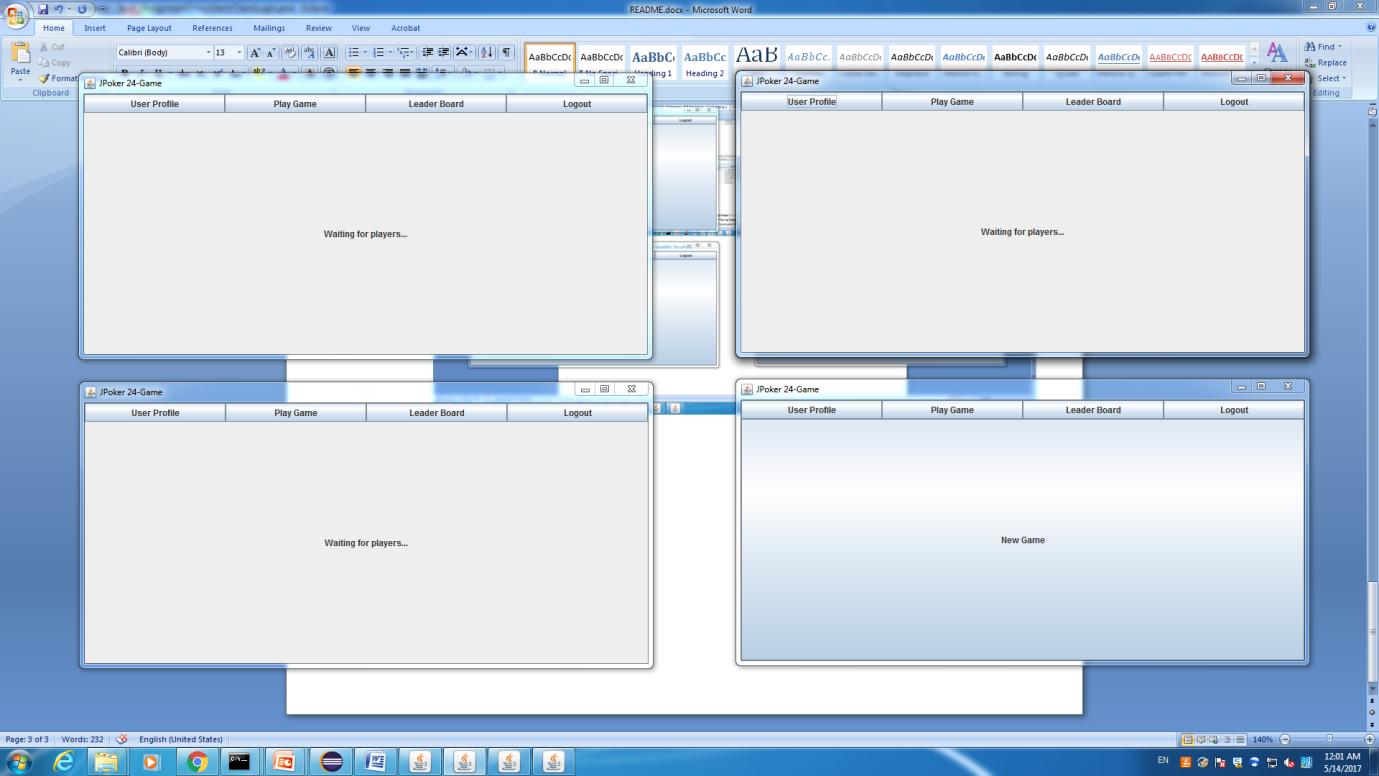


Here are the leader boards.

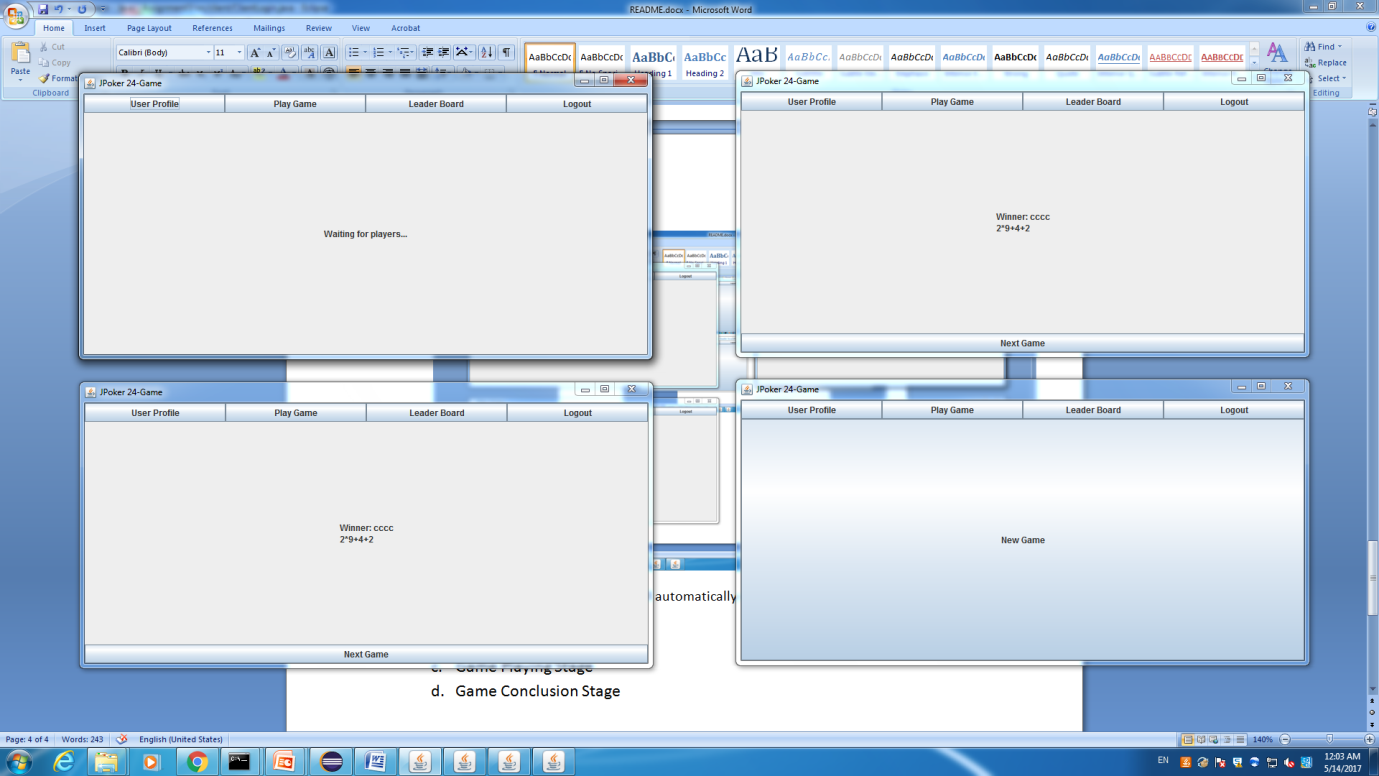


Ready for begin.

1. Waiting Stage

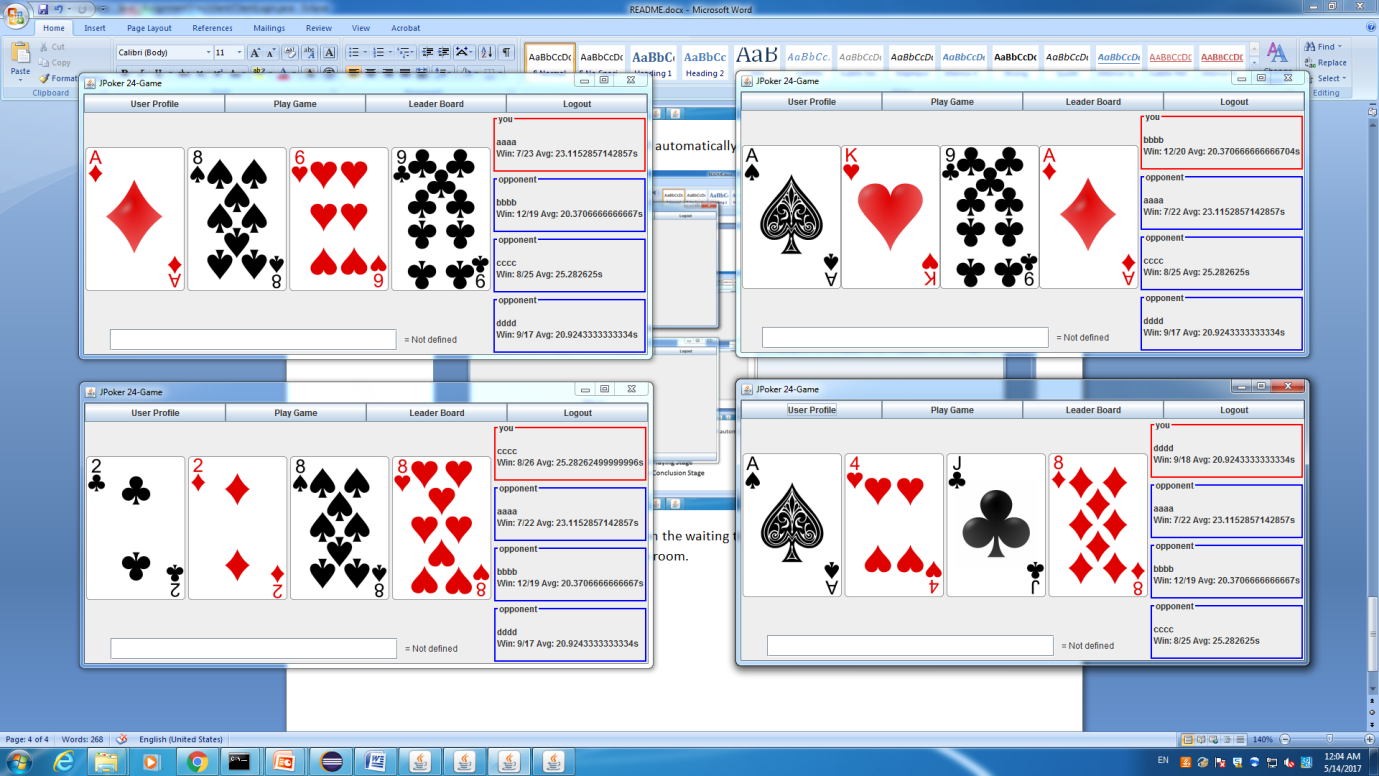


There are 3 clients waiting, game will automatically start in 10s.

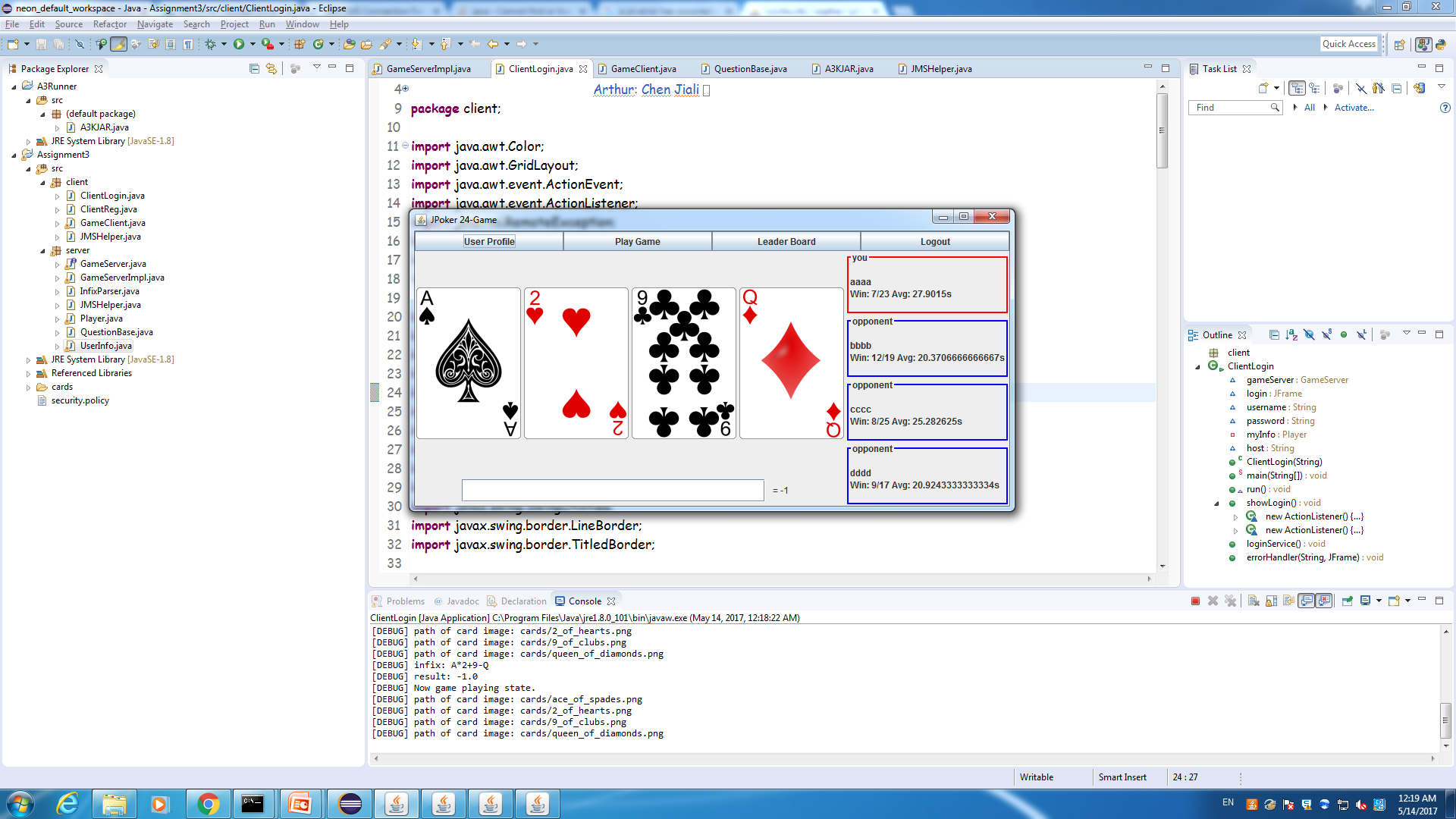


There is only one client waiting. When the waiting time exceeds 10s, the game will start as soon as the second player enters the room.

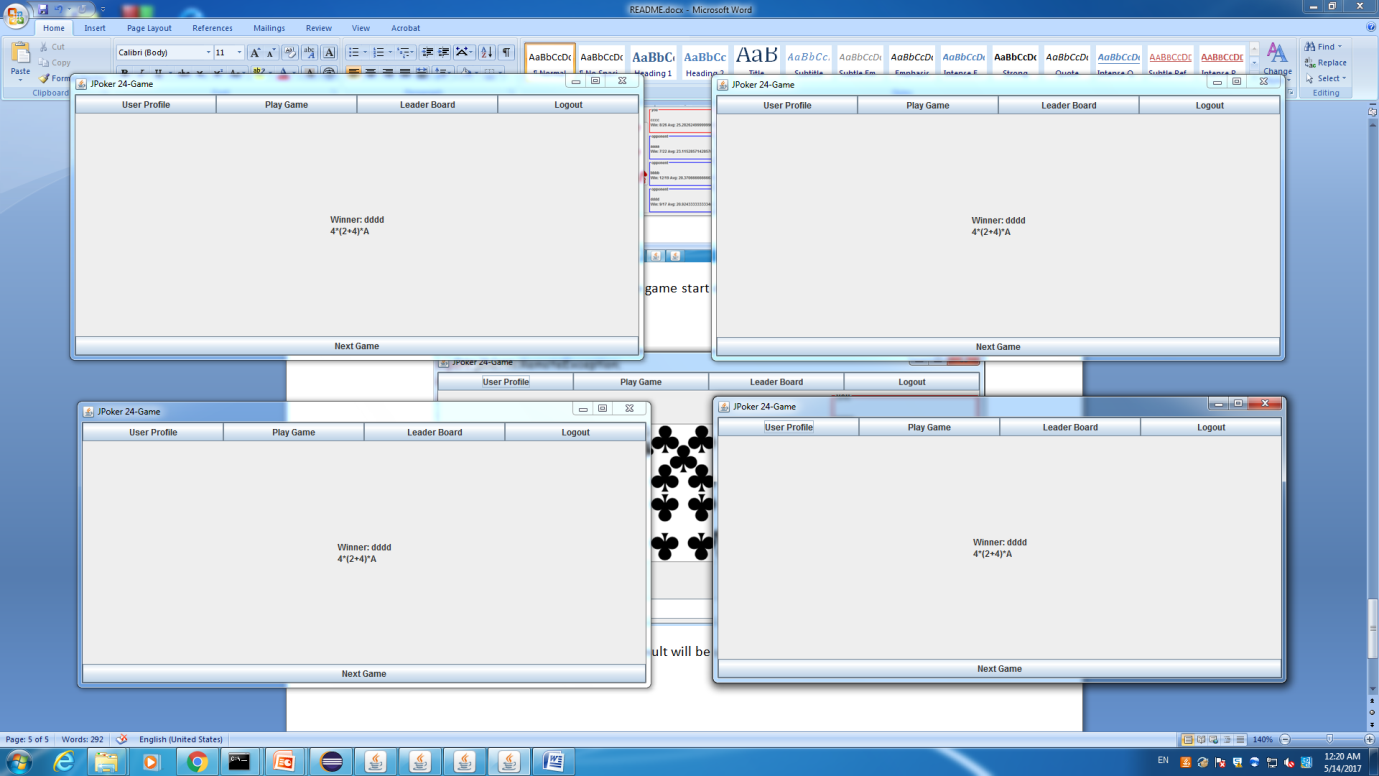
1. Game Playing Stage



With four client start the game, the game start directly.

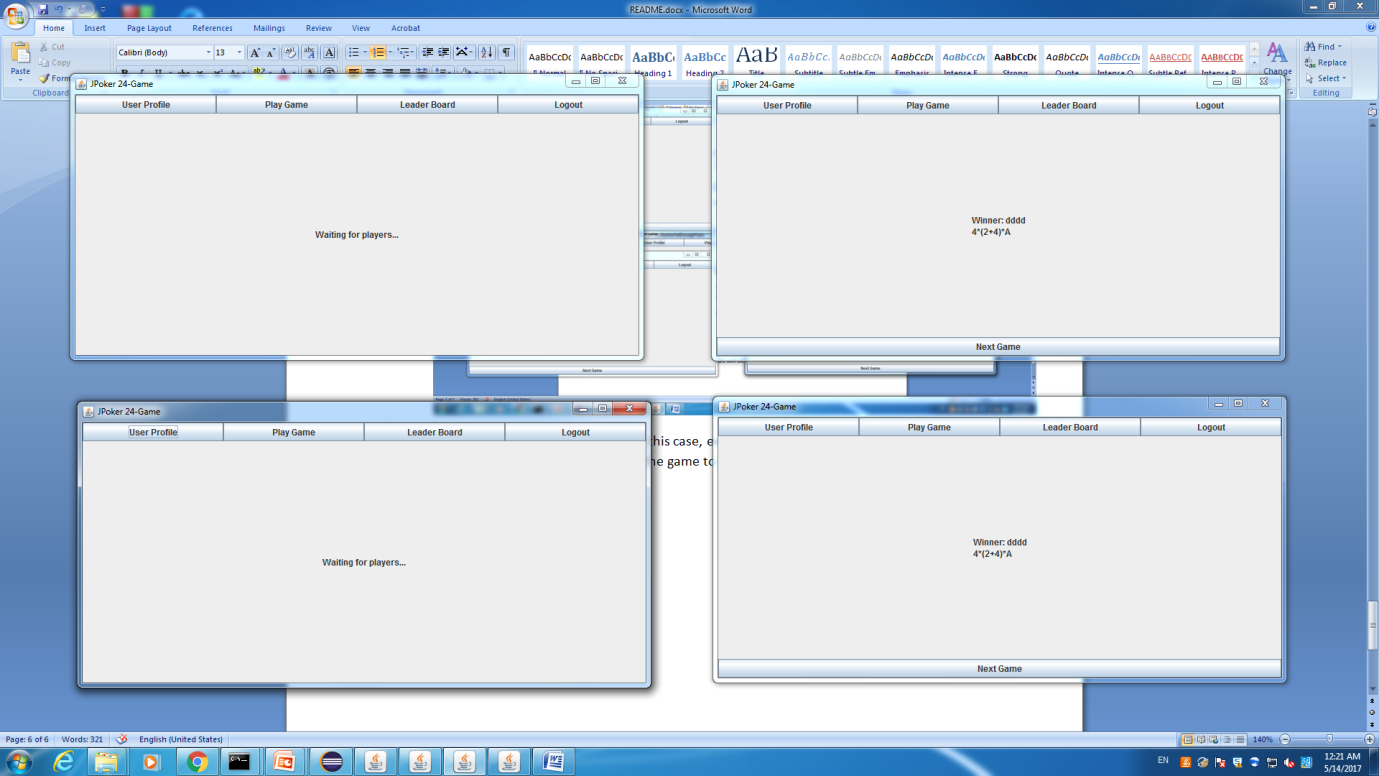


When one got wrong answer, the result will be updated immediately and game continues.

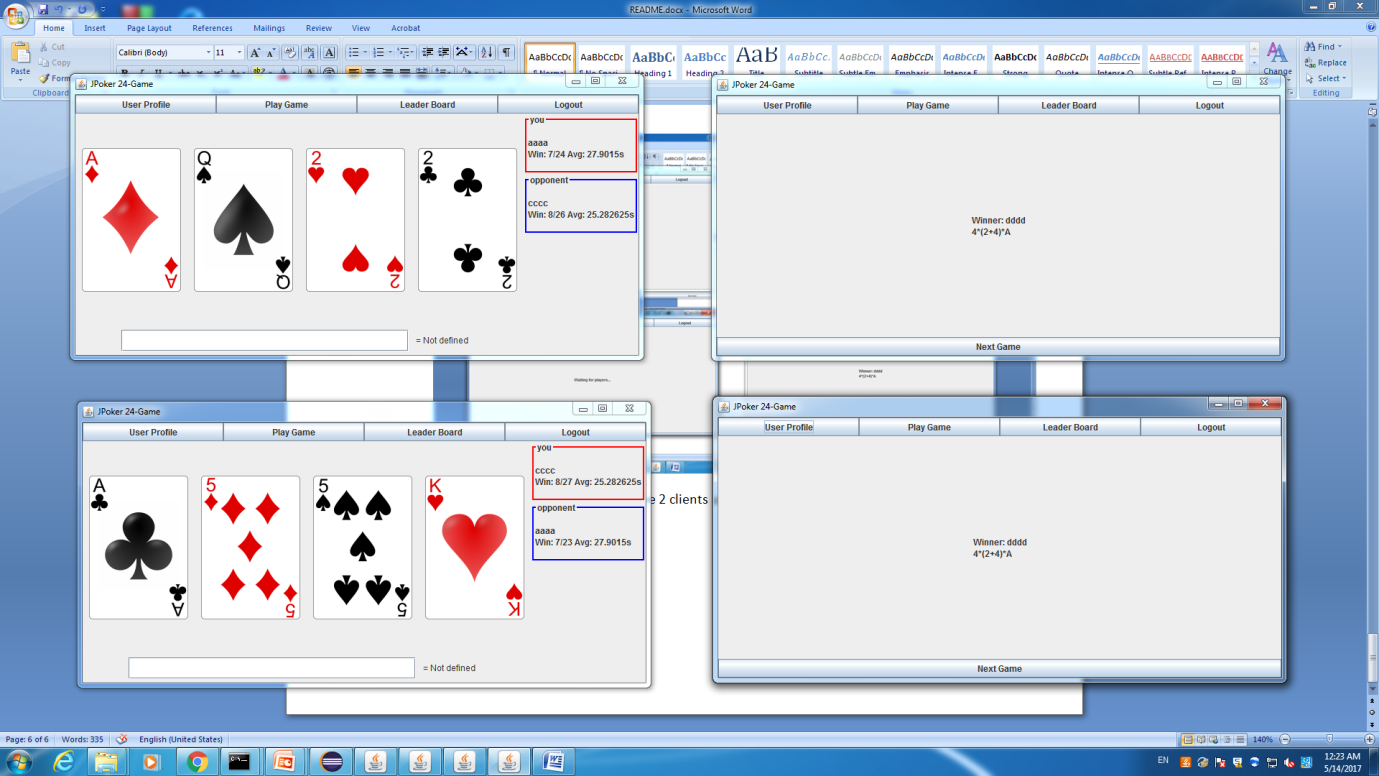


When there is a winner, say dddd in this case, everyone will be noticed by the server information of the winner and stop the game to game conclusion stage.

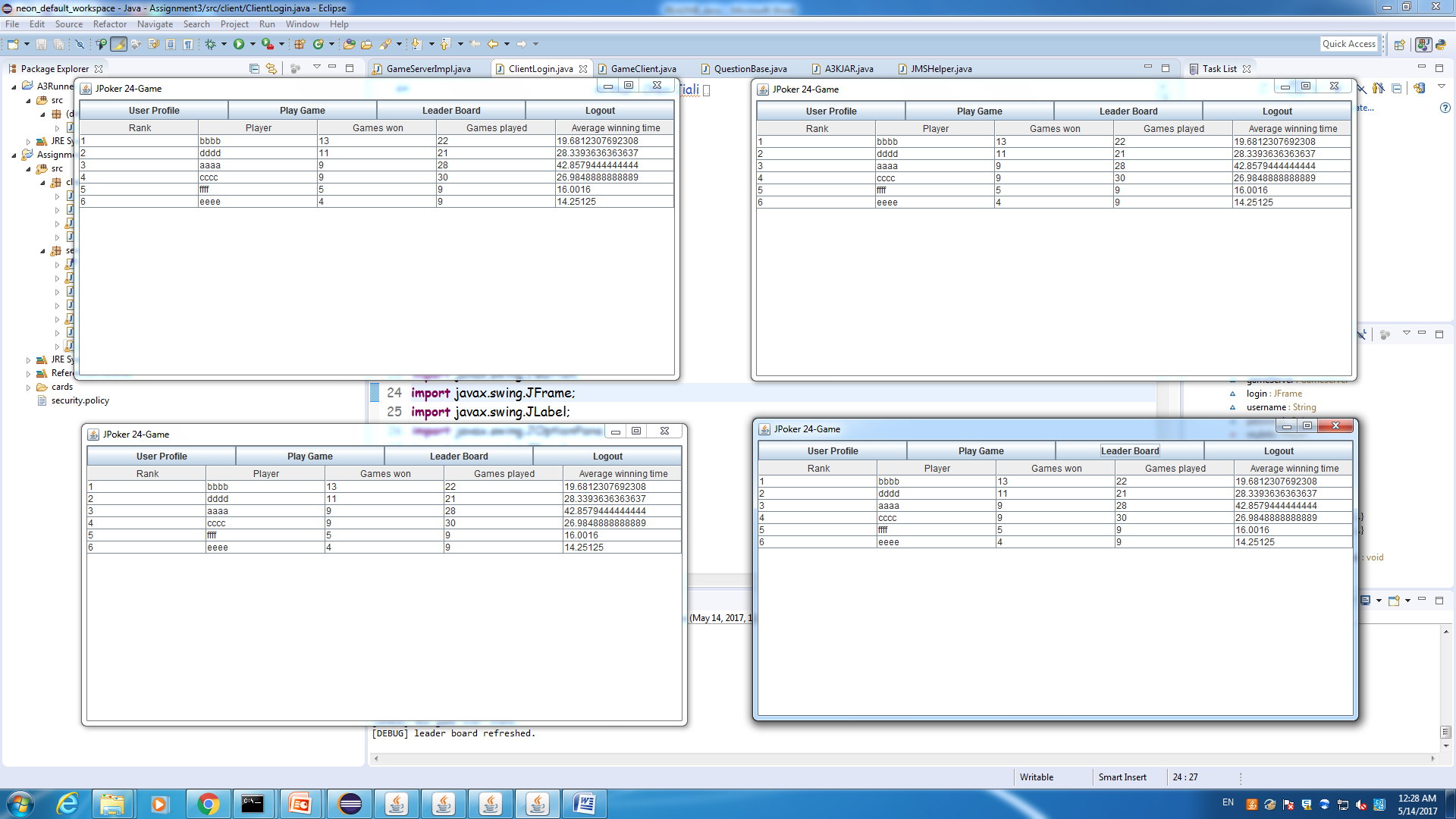
1. Game Conclusion Stage



The logic is similar to initial part. Here 2 clients ask for next round.



The one-versus-one game starts.



After several rounds, you can easily see the updating of the data.