

Assignment 3, SDJ2

(MVVM, RMI)

The assignment:

You must re-design and re-implement the chat system from Assignment 2, this time using RMI instead of Sockets.

You *may* completely redo the assignment, but the idea is that you take your assignment 2 and just delete the sockets part (layer) and insert RMI instead, leaving most of the assignment untouched.

We have talked about keeping the layers in your system separate, and loosely coupled. This assignment should show you, if anything *outside* of the networking layer is using anything from *inside* your networking layer, meaning you do not have a clear separation.

Some of you have not kept this separation strict and may find it troublesome. Reflect upon what could have been improved in your assignment 2 regarding architecture.

Requirements

- The application must use RMI for the clients connecting and sending/receiving messages (method calls, the sockets in Assignment 2 being replaced by RMI)
- You must follow the MVVM pattern and use the Observer pattern.
- It is required to make a class diagram for the final solution. In the diagram, you must be able to identify the MVVM parts and the design of the RMI related parts. If other patterns are used, then these also have to be clearly marked with notes.

Deadline

See itslearning.

Format

It is ok to work in groups. Hand in a single zip-file with

- Class diagram
- Source code for all Java classes (zip the src folder)
- Related resources like fxml files, and if used, external jar files

Evaluation

Your hand-in will be registered and counts for one of the exam requirements.