Faculty of Computers, Informatics and Microelectronics

Technical University of Moldova

APPOO

Laboratory work#4

System design.

Author:
Denis POSTICA

Supervisor:
Anastasia SERSUN

May 26, 2017

Laboratory work #4

1 Purpose of the laboratory work

Assignment 4 will partially cover your individual Project. It will focus on building the diagrams for your system. Diagrams are important way of communication to other team mates, or even to other teams and companies about your system. Sometimes it's easier to draw than to describe in words. That's why we're using pictures and icons instead of words.

As diagrams meant to ease the communication, during this assignment you will communicate! You should choose a pair, that will be the one who will try to understand the structure of your project, and after that will present it!

Each project can be described from different points of view. You will use 4 UML diagrams to describe your project:

- Use-case diagram.
- Activity diagram.
- Collaboration diagram / Sequence diagram.
- Component diagram.

This will ensure more deep understanding of your system.

2 Laboratory work implementation

As my Course Thesis I chose to implement a group chat using Java Sockets. The connection and data transfer is provided by the TCP. There is one server that waits for Clients to connect, when a Client connects, a separate Thread is created for handling commands, messages from that Client. Available commands to the Client are: 'help' - prints list of all available commands; 'list' - prints the list of users on-line; send a message to other users on-line (by default if the user input isn't one of commands it's a message); 'exit' - used to exit the chat. The code of the program can be found at Git repository `https://github.com/PosticaDenis/APPOO`.

Laboratory work analysis

USE CASE DIAGRAM

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). Each use case should provide some observable and valuable result to the actors or other stakeholders of the system.

Usually use case diagram summarizes who uses your application or system, and what they can do with it.

In the following diagram I represented some actions that a Client can perform using chat application(on the right); also, the diagram for all the necessary steps to perform a certain action within application (those steps are marked with ¡¡include¿¿), on the other hand there are also steps which are not required, but also wouldn't affect the desired action if being executed at some moment in the time before ending the action (those steps are marked with ¡¡extends¿¿).

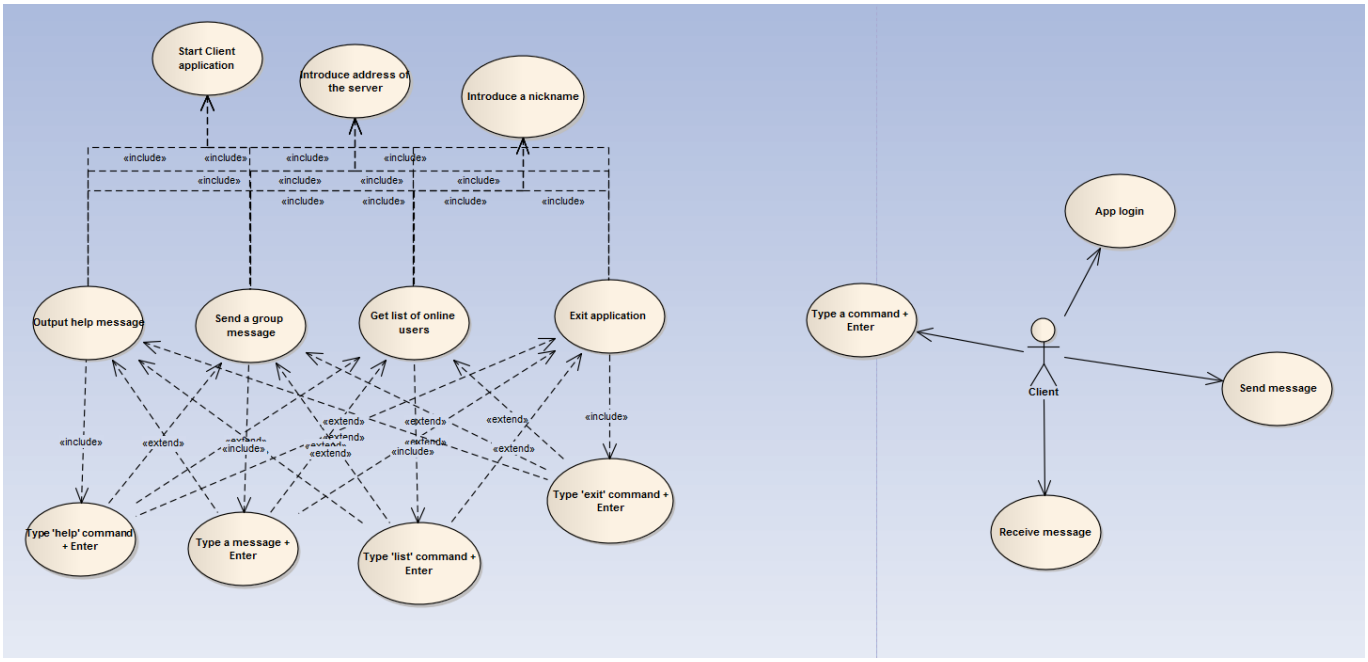

Figure 2.1 – Use-case diagram for some actions in the chat.

ACTIVITY DIAGRAM

An activity diagram shows a business process or a software process as a flow of work through a series of actions. People, software components, or computers can perform these actions. You can use an activity diagram to describe processes of several types, such as the following examples:

- A business process or a flow of work between users and your system.
- The steps performed in a use case.
- A software protocol, that is, the permitted sequences of interactions between components.
- A software algorithm.

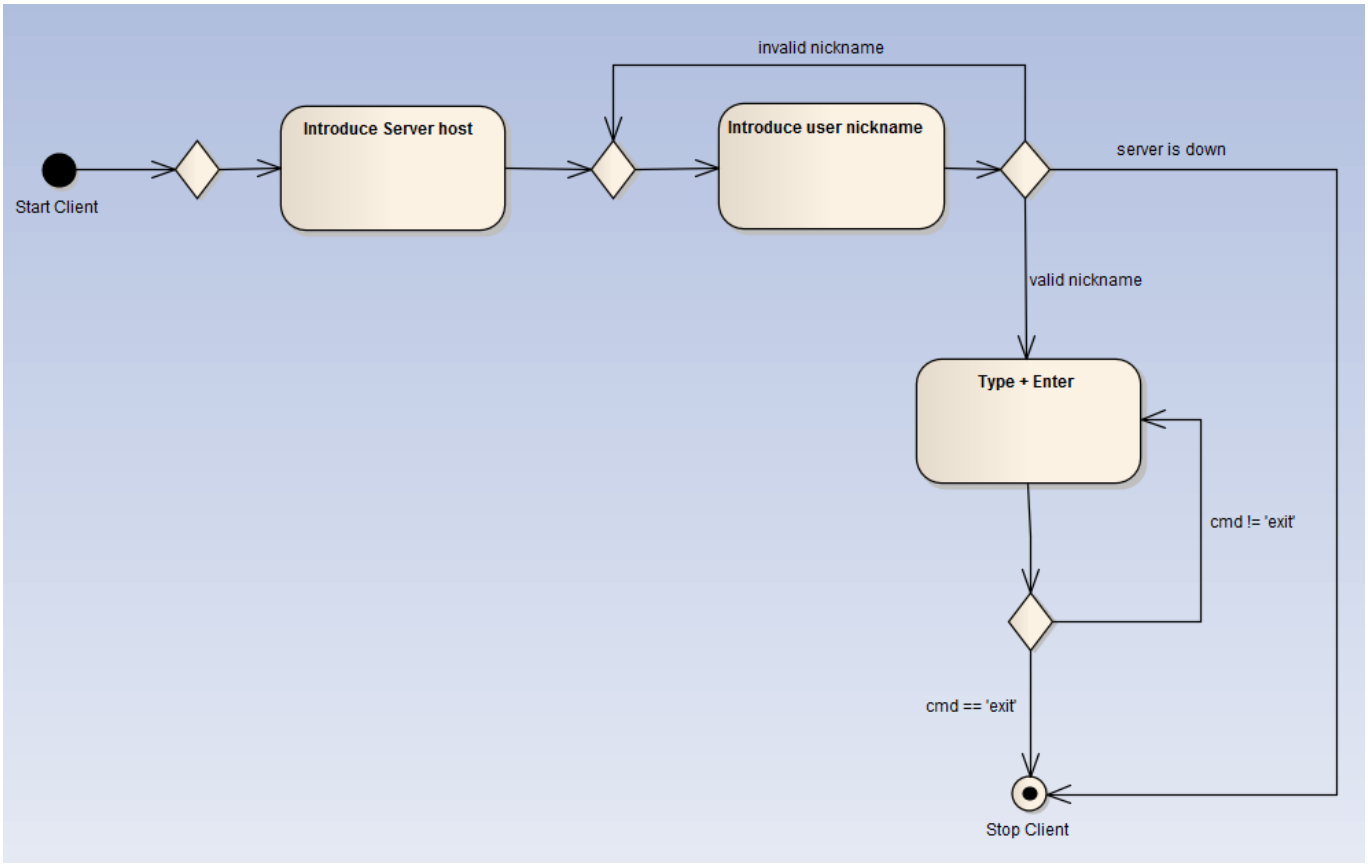

Figure 2.2– Activity diagram for actions in the chat.

SEQUENCE DIAGRAM

Collaboration and Sequence diagrams represents the same information in different ways. Collaboration (communication) diagram shows interactions between objects and/or parts (represented as lifelines) using sequenced messages in a free-form arrangement. Oppoisite, Sequence diagram shows message flow in more structural way.

Communication diagram corresponds (i.e. could be converted to/from or replaced by) to a simple sequence diagram without structuring mechanisms such as interaction uses and combined fragments. It is also assumed that message overtaking (i.e., the order of the receptions are different from the order of sending of a given set of messages) will not take place or is irrelevant.

You should present only one of them, decision is up to you.

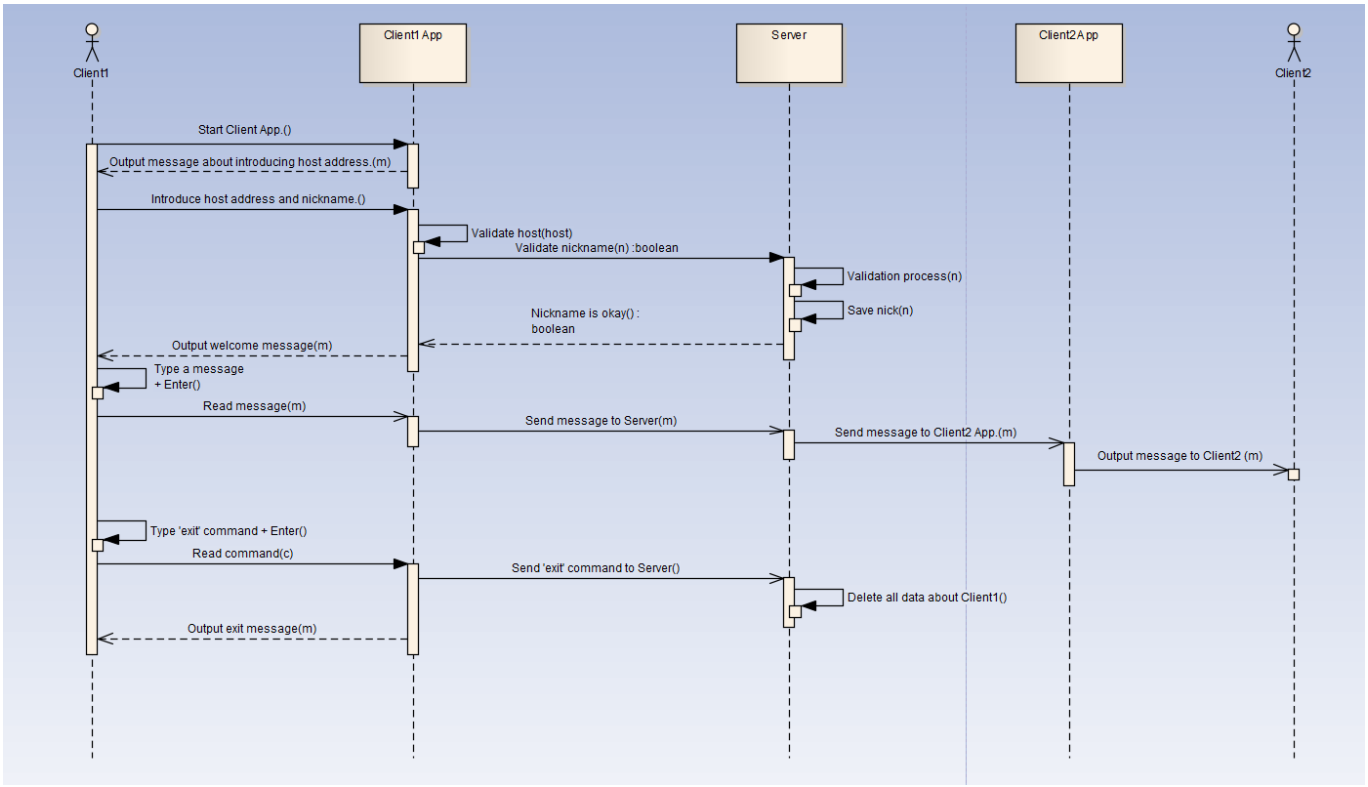

Figure 2.3– Sequence diagram for sending a message in the chat.

COMPONENT DIAGRAM

A component diagram shows the parts of a design for a software system. Component is a reusable piece of system functionality. A component diagram helps you visualize the high-level structure of the system and the service behavior that those pieces provide and consume through interfaces.

The artifacts that implement component are intended to be capable of being deployed and re-deployed independently, for instance to update an existing system.

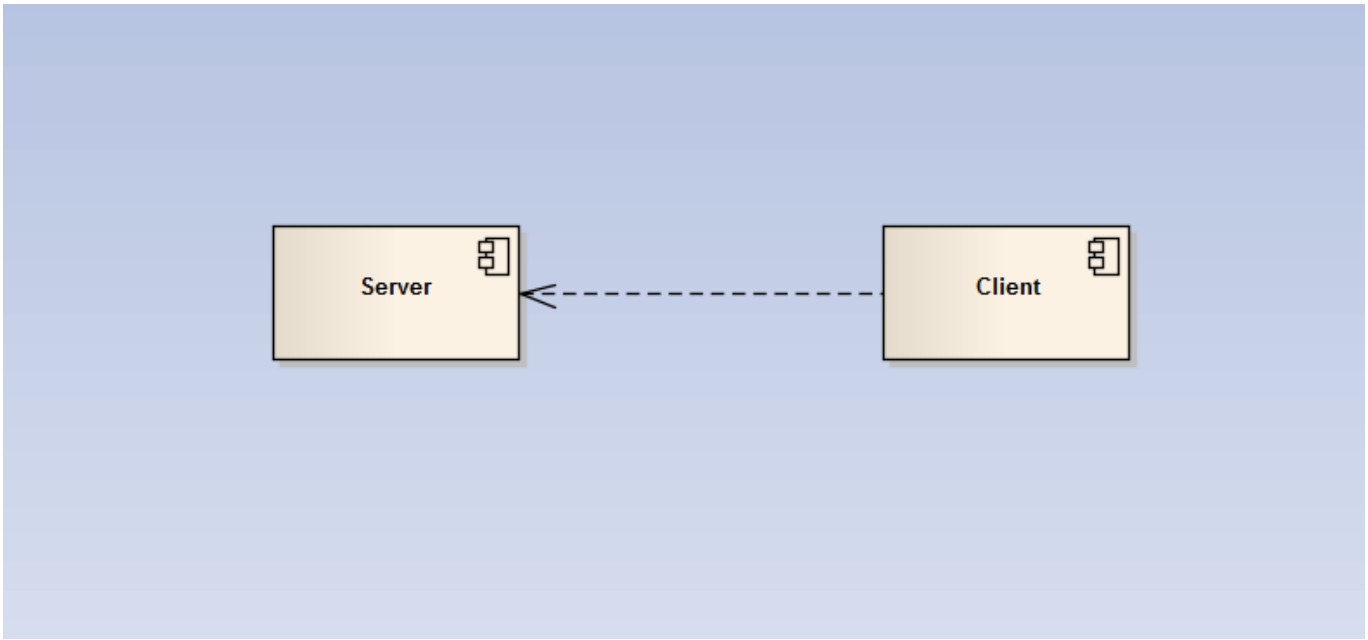

Figure 2.4– Component diagram of the chat.

Conclusions

References

1 Guide for laboratory work 4.