PARALLEL AND CLOUD COMPUTING

LAB ASSIGNMENTS

ASSIGNMENT 4

Deadline: Monday 29 April 2019, 23:55pm

Lab sessions: Wednesdays, 10:20am–12:10pm

Location: Room 406, Lichee Hill 6

Lab Instructors: Christos Markos (11760007@mail.sustc.edu.cn),
刘健 Jack (11849061@mail.sustc.edu.cn),
华正昌 Pill (11510644@mail.sustc.edu.cn).

DESIGN AND IMPLEMENTATION OF A PUBLISH-SUBSCRIBE SYSTEM USING ACTIVEMQ

INTRODUCTION

You are required to demonstrate working programs and to submit well-commented code and a lab report. Any text should be in English only.

PREAMBLE

Before working on this assignment, you should master JMS and ActiveMQ programming. Here is a list of helpful materials, which can be found on the course's Sakai page under the *Labs* section:

- *JMS & ActiveMQ.pdf* (introduction to JMS and ActiveMQ);
- *ActiveMQ In Action* (textbook by Snyder. B., Davies, R., and Bosanac, D.).

EXERCISE SPECIFICATION

For this exercise, you are asked to design and realize a publish-subscribe system using ActiveMQ, based on Assignments 1, 2, and 3. Such systems are used in a wide variety of application domains, particularly those related to large-scale dissemination of events. You are asked to design a knowledge sharing platform system using the publish-subscribe model; examples include Quora, ZhiHu, Brainly, and so on. You do not need to implement the part about questions and answers (e.g. in Quora), just provide a way to share knowledge and experience on one topic, and for those who are interested in the topic to receive what others share.

Based on your own optimized RMI framework and register-login system, you will add more features to realize this system:

1. When a user is registering in the system, the system will guide them to choose (subscribe to) some topics that they are interested in from a given topic list.

2. When a user logs in, they will receive messages about their subscribed topics. They can also publish messages on a topic and these messages should be received by those users who have already subscribed to this topic.
3. You can realize either non-durable or durable subscription. You will receive more marks for the latter.

The following is based on the specification of Assignment 1. In red is the additional client functionality required for Assignment 4. On the server side, you need to install ActiveMQ and start the Broker.

Register Operation:

Client: The client should guide the user to input his username and password, then choose the topics that he is interested in, and then invoke a remote object to store the new user at the server site.

Server: The server will store the new user information and acknowledge the success of the operation.

Login Operation:

Client: The client should guide the user to input his username and password, then invoke a remote object to check the validity of the provided credentials. If login succeeds, the client gets the user's subscribed topics list from the server, sets up a connection with the Broker, and begins to receive and show messages relevant to the topics.

Server: The server implements the object that checks the validity of the credentials (username and password) received by the client and acts accordingly: it either accepts and acknowledges it or rejects the request and generates an error message to the user at the client side. The user information should be encoded.

Publish Operation:

Client: If login succeeds, the client can provide a publish option. The user can choose a topic and write something about the topic, and then publish the message.

SUBMISSIONS

Submit your assignment named <studentID>.zip with your code and report to the course's Sakai page. Submissions by e-mail or other means will not be accepted. Make sure to submit your assignment by the set deadline.

ASSESSMENT

The full marks for this this exercise are 100 and they are distributed as follows:

- System implementation: 70%
- Report: 30%

System Implementation:

The key aspects you should focus on are:

1. Implementation of basic publish-subscribe model: a publish client can publish some message on one topic, and a subscribe client can receive information of interest.
2. A user can be both a publisher and a subscriber at the same time.
3. Utilizing durable subscription in ActiveMQ, users will not miss subscribed messages even when offline.

Functions	
Basic publish-subscribe model	30 marks
Client can be publisher and subscriber	20 marks
Durable subscription	10 marks
Total	**60 marks**

Quality of the code (structure, layout, comments): 10 marks

Report:

Design	10 marks
Problems	5 marks
Demonstration	5 marks
Quality (layout, structure, language)	10 marks
Total	**30 marks**

The template for the report is provided in the next pages.

PARALLEL AND CLOUD COMPUTING

REPORT

LAB ASSIGNMENT: 4

Student Name: _____

Student ID: _____

Student E-mail: _____

DESIGN

Describe the design of your system by providing the following information:

- Module design: structure block - flow chart and description
- Class design: UML class diagram and description

PROBLEMS

Describe major design decisions you had to make and the problems you encountered when implementing the system together with the solutions you devised.

RUNNING RESULT

Provide an illustration of your running system by providing screenshots of all the different steps and dialogues and any other relevant information you consider important.