

CMPE 273 – Enterprise Distributed Systems

Lab 2 – RabbitMQ and MongoDB

Due: 17 April 2016

This lab covers designing Message oriented application using RabbitMQ. This lab is graded based on 20 points and is an individual effort (no teamwork allowed)

Prerequisites:

- You must have carefully read the environment setup document. You should be able to run RabbitMQ sample example
- You should have prior knowledge of JavaScript, default Sessions

The Assignment

- You will be developing a client and server
- **Use MongoDB as the database. Sessions should be stored in MongoDB.**
- **Passwords need to be encrypted.**
- On, or before the due date, you have to turn in the following:
 - Code listing of client and server
 - Document with architecture of the RabbitMQ interaction in your client/server application, system design description and screenshots

Grading

This assignment will be graded on the following points:

- 0 – 20 based on the correctness of your solution

Late assignments will be accepted, but will be subject to a penalty of -5 points per day late:

Submissions shall include source codes for each client/server pair, plus screen image captures of each client/server pair during and after run. (Source codes should be submitted by canvas)

Part 1: Twitter Application (17 Points)

Server (8 Points)

Server should perform following tasks:

1. Basic User functionalities:

- a. Sign up the new users (First name, Last name, Email, Password). Passwords have to be encrypted
- b. Sign in with existing users
- c. Sign out

2. Profile:

- a. About: Birthday, Twitter handle, contact information and location

- b. Followers and Following list. You should be able to follow people.
 - c. Show your tweets and re-tweets
3. **Twitter feed functionality** showing tweets of people you are following and option to re-tweet.
4. **Implementing Hashtag (#)** functionality (in Search and Tweets).
5. Implement **Connection pooling** for database access

Validation is extremely important; exception handling should be implemented. **Implementations with proper validations and exception handling will attract good marks.**

Client (6 Points)

Client must include all the functionalities implemented by the web services. Develop the Client using HTML5 and AngularJS and Bootstrap. **Client similar to Twitter will attract good marks.**

Testing of the server should be done using JMeter and Mocha. Mocha is a node.js testing framework.

Following tasks to be tested using JMeter: (2 Points)

Test the server for **100, 200, 300, 400 and 500 concurrent users** with and without RabbitMQ. **Draw the graph with the average time and include it in the report.**

Following tasks to be tested using Mocha: (1 Point)

Implement 5 randomly selected REST web service API calls using Mocha. **Display the output in the report.**

Part 2: Questions (3 Points)

1. Explain what performance change RabbitMQ provides? Elaborate on the results of throughput with and without using RabbitMQ. If you find any increase/decrease in the throughput, explain the reason for the same.
2. Explain the strategy used in implementing Sessions in this Lab. Compare your Sessions strategy with default express Sessions. Describe which Strategy is better.
3. If given an option to implement MySQL and MongoDB both in your application, specify which data of the applications will you store in MongoDB and MySQL respectively

Deliverables Required:

- Submissions shall include source code only for each client/server pair
- Project directory must include the group ID/Name (e.g., Lab1-cafeine)
- Archive the project, and report into one archive file (e.g., zip)

- Do not submit binaries, .class files, or supporting libraries (e.g., junit.jar, javaee.jar) (including them would be 3points deduction).
- Project report
 1. Introduction: state your goals, purpose of system,
 2. System Design: Describe your chosen system design
 3. Results: Screen image captures of each client/server pair during and after run.
 4. Performance: What was performance? Analyze results and explain why are you
 5. getting those results.
 6. The answers to the questions posed

For example: Smith is submitting a project. You have provided the following files and source directory:

- smith-lab2-report.doc
- lab2/ (do not send class or jar files)

```
zip -r Lab2-smith.zip Lab2
```

Submission

- On-line submission: shall include all source zipped with your last names (ex. Lab2-Smith.zip) and your report (smith_lab2_report.doc). Submissions shall be made via Canvas.