COMP 3670 - Fall 2020 Computer Networks School of CS

Assignment 02

(Worth 10% - Due date October 25, 2020)

Bill Shema 104822276

Ariya Rasekh 105127455

Baljot Hansi 105160856

# Instructions

This assignment is divide into two parts. Part 1 is short questions; this part is an individual assignment. Part 1 is NOT a group assignment. Part 2 is a design and coding assignment. Part 2 is a group assignment.

**For submission -** Each student should submit his answers to part 1, using the blackboard assignment submission feature. For part 2, each student should submit one page detailed his role and responsibilities within the group (e.g. which task (design and or coding) completed by him/her). In addition, include in this page a link to a version control repository (e.g. GitHub, BitBucket) contains the design document(s) and the source code of his group.

# Design and Programming Assignment [70 points]

You are required to design and implement a communication protocol for a network application. The network application enables a group of nodes (end systems) to communicate over TCP/IP networks. The nodes in this network application are divided into two groups, namely job-seeker and job-creator. For simplicity, assume that the node during any active communication scenario could only play one role (job-seeker or job-creator). Also, the job-seeker nodes do not connect directly to each others. Each job-seeker connects to one job-creator at a time, even if there are more than one job-creator. The job-seeker offers his services (e.g., computational resources) to the job-creator. The job-creator depends on the availability of a job (e.g. computational task). It could assign the job immediately to the job-seeker or take a note of the job-seeker availability to give it a job when a job becomes available. There is a predefined set of job-types that the job-creator could assign to the seekers. Depend on the job size, the creator could give the same job to multiple job-seekers. After the job is assigned to the seeker, the connection could be maintained or terminated depending on the job. When the seeker completed the job, it reports the job status and any result to the job creator. The creator could assign multiple jobs to the jobseeker, but one job at a time. The job seeker could accept/reject a job. The creator or the seeker could choose to terminate the connection at any time.

Q1) Design a network application protocol for this application and justify your design decision. In your protocol design, you should address the following. *[35 points]*

1. The communication patterns of the network application.

The communication is one-to-one and its written in Java. The pattern is: server (job creator) must run first, then the job seeker, and these two nodes will be connected, then based on the messages, jobcreator or jobseeker execute different reactions.

1. Define the protocol design goals.

This is client-server model, with a one-to-one connection. The server remains in contact with the client throughout the whole interaction(session). At a beginner phase, both the server and one client remain connected throughout the whole session. In our case jobcreator acts a server while jobseeker acts as a client.

1. Define the message format, structure and semantics.

The message format is String, the structure is that the job creator sends a text to the jobseeker, offering different types of jobs to the jobseeker. Then jobseeker can accept or refuse by sending a text to the job creator. based on the received text, the job creator decided what to do. If job seeker accepts, then the job seekers get a job report, otherwise, it offers again. Also, job creator can get a message from the job seeker which would be informed that job seeker is busy, and he/she can be free in certain amount of time.

1. Design the communication rules.

The jobcreator can only interact with only one jobseeker.

One jobseeker can have multiple requests to the jobcreator.

Q2) Argue the need for a new application layer protocol for this network application instead of using existing standard protocols (e.g. HTTP, SMTP, WebSocket, etc.) [10 points]

Existing network protocols are important for foreign application to be able to interact. For example, without the use of HTTP, the world wide web wouldn’t be as accessible as it is today. If every email provider had their own protocol, email communication would be extremely hard between two different email providers but since they all use the smtp protocol, access to emails has never been easier. Having a unique protocol is good for the common efficiency of its good. We want our protocol to be very efficient, hence the need for a new and moderated protocol.

Q3) Provide the implication source code of your network application protocol with sufficient test cases based on the design goals, message philosophy (format, structure, semantics), and Communication rule. [*25 points*]

Please find the source code on GitHub.

Link: https://github.com/nohack11/comp3670-project