Assignment 4 Dry Part

Raphael Astrow, 922130174, rastrow@andrew.cmu.edu
David Sainz, 927902023, dsainz@cs.technion.ac.il

Special notes on how to run:

This must be run on JDK 7.0 in order to use the java.nio.file dependency. The thread sizes and base path are changed in the config text file while the type handlers are changed in the config xml file.

Short description of your implementation. Include in this description the following details: General implementation notes - a summarized external documentation, covering design aspects.

Our implementation follows assignment suggestions. Connection listening happens in WebServer.java, SocketReader.java is our socket reader threads, Request.java is our request object, RequestHandlerThread.java is our request handler threads and TSP Engine implements TypeHandler. The request handler threads initiate sessions and use SessionManager to keep track of all existing Sessions in the system. Threads are processed one by one from The LinkedBlockingQueues RequestQueue and SocketReader.

What is the value you set, in your server, for session timeout?

300000

What happen if you use 0 (unlimited timeout)? What happen if you use a short timeout? a long timeout?

A short timeout makes a session be updated frequently – every time the session expires – which would mean that more client requests would result in updated cookies. A long timeout makes the session be updated less frequently. In our implementation, an unlimited timeout makes a session be updated for every request.

2. In this exercise you were asked to implement session management on memory. What are the disadvantages of this approach? Offer a solution that overcomes these problems.

One disadvantage of this system is that is memory intensive and when the system is overloaded with many client requests – a lot of the system's memory will be reserved for managing sessions. As a result, the system might be slower in performing its other tasks

once sessions start eating up lots of memory. A persistent storage system – perhaps a Cache like we implemented for assignment 3 would help overcome some of the in-memory problems. Every time a request is made, a session can be looked up in the database by its UUID and retrieved to the system. This way, hard disk space and not system memory will be more responsible for session management.