CS5300 Project1b

Kaitlyn Kwan (kyk25), Matthew Tomlinson (mat297), and Sam Nelson (sln58)

3/26/15

Session.java

Contains a Session object, which has sessionID, version, message, and expirationTimeStamp.

It also has a constructor that parses SessionWrite Requests.

Server.java

Contains SessionRead, SessionWrite, and ExchangeViews

SessionRead basically follows the outlined code given in the write-up. If an IOException occurs, it tries to send the datagrampacket once more. It sends a datagrampacket with the format [callID, operationSESSIONREAD, sessionID]. The server that receives this datagram would than call the function ResponseToSessionRead. It sends back a response with the format [callID, “found” session Message, session ExpirationTimeStamp] or [callID, “not found”], depending on if the session in its sessionTable.

SessionWrite basically follows the SessionRead, but sends a datagrampacket with the format [ callID, operationSESSIONWRITE, sessionID, version, expirationTime, data ]. The server that receives this datagram would than call the function ResponseToSessionWrite. It sends back a response with the format [ callID ].

Project1a.java

This is the file from project1a with the necessary added code for Project1b.

Cookie format: [sessionID, versionNumber, primary server, backup server]

Determines if there is an existing cookie. If not or the client has logged out, a new session is created and added to the sessionTable. The session contains default perameters of “Hello User”, 60 expiration time, unique sessionID, and version 1. The given server then attempts to backup this new information to some view with SessionWrite. If not possible, backupServer is null. A new cookie is created and added to the response.

If it is a returning client, the server checks if it is the primary or backup server. If so, it retrieves the session from the cookie. If not, the server sends a SessionRead to the primary server – or backup server, if primary is not available – to get the necessary information. It takes the information and adds an updated version to the given server’s sessionTable. It updates the version, expiration time, and message (if message changed). It then sends a SessionWrite to backup this new information. A new cookie is created and added to the response.