LAB-2 Write-Up

Shubham Renukadas Simant

1001860599

srs0599@mavs.uta.edu

Steps to Follow:

1. Run serverB in command line using py serverB.py first.
2. Run serverA in command line using py serverA.py.
3. Run client in command line using py client.py.

Execution Process:

* When serverB starts running it waits for client i.e serverA to make request.
* When serverA starts running it connects to serverB and wait for client to request data.
* When we run client, client send request to serverA, serverA collects the data from serverB.
* ServerA combines it’s data with collected data of serverB.
* ServerA sorts the data on basis of file name.
* ServerA send the data to client.
* Clients display the sorted list of files present in directory\_a and directory\_b.

Requirements:

* Watchdog.
* handler
* Dirsync
* hurry.filesize
* socket
* pickle
* traceback
* datetime

following things need to be implemented

Citations:

<https://www.youtube.com/watch?v=3QiPPX-KeSc> – Socket client-server connection.

<https://www.edureka.co/blog/socket-programming-python/> - Socket client-server connection.

<https://flaviocopes.com/python-get-file-details/> - Files Meta data

<https://www.geeksforgeeks.org/python-sort-python-dictionaries-by-key-or-value/> - Merging the file details of serverA and serverB.

<https://stackoverflow.com/questions/17330139/python-printing-a-dictionary-as-a-horizontal-table-with-headers> - Formatting sorted list of files at client side.

<https://www.instructables.com/Syncing-Folders-With-Python/> - Sync files in directories of Server A and B.

<https://stackoverflow.com/questions/37227176/how-to-send-a-message-from-client-to-server-in-python> - Keeping two way communication between client and server.

<https://michaelcho.me/article/using-pythons-watchdog-to-monitor-changes-to-a-directory> - Watch Dog Watcher and Handler

<http://thepythoncorner.com/dev/how-to-create-a-watchdog-in-python-to-look-for-filesystem-changes/> - Watch Dog Implementation