CSE 5306 – 001- DISTRIBUTED SYSTEMS

REPORT

GAURAV CHAJJED MAHAVEER – 1001861652

SHYAM JIVANDAS RABADIYA - 1001873983

I have neither given nor received unauthorized assistance on this work.

Signed: **Gaurav Chajjed Mahaveer, Shyam Jivandas Rabadiya**

Date: **03-03-2022**

Programming Language used is PYTHON

The above Project is divided into four parts were,

Part 1- Create a virtual machine and install Linux. Using Ubuntu environment to test the program

Part 2- Implementing Multi-threaded file server to Upload, Delete, Rename and Download file

Part 3- Synchronized Storage

Part 4- Synchronous, Asynchronous and Deferred Synchronous RPC

EXPLANATION

Part 1: Virtual Machine –

In this part of the project, we installed the LINUX operating system on our Windows 11 Desktop and set up the environment to run the program on Ubuntu distribution.

Part 2: Implementing Multi-threaded file server

To implement multi-threading in python we use py thread class WorkerThread. In the server file we create a loop where we give all the file operations such as Upload, Download, Rename and Delete. We create gaurav.txt file in our folder and use that file to do all the file operations. When we give Upload command on the client terminal it goes to the socket and tracks the path and uploads the file in the folder by creating an upload directory. We can open multiple terminals and run the client on the LINUX. The other operations also follow the same rule.

Part 3- Synchronized Storage

We have synchronized client side with the server side i.e. directory_a with directory_b which is automatically handled by a helper thread. We have checked and compared the common files in both those directories. We sort the file list by file name and get the duplicate of files present in both the directories as one of the files is locked (lock/unlock operations). We also compared the modified time between the two files. This way we have removed the oldest modified file from the directory. Then we sent the details of all files in a sorted way present in directory_a and directory_b to the client. Then we have represented the result like file name, file size, modified date, file content, etc. in client.py file in 2-d form. Thus, if the content of a file is updated, the entire file is sent to the server to overwrite the original copy at the server. In the other words, we can conclude that any operational change (i.e. creating a file, deleting any file, updating the content of any file, renaming any file name, etc.) done on client side will be reflected to the server side in the real time frame.

Part 4- Synchronous, Asynchronous and Deferred Synchronous RPC

This part is divided into two different parts

Synchronous RPC: To implement synchronous RPC we use the TCP protocol and implement the server-client communication. We have implemented two functions i.e. add(i, j) and sort(array). We have used the JSON package from python to establish the communication.

First, we start the server terminal, once the connection is established, we then run the client on different terminal and perform the add and sort function.

Asynchronous and Deferred Synchronous RPC: Here we have used JSON package to send and receive data from client and server. We have implemented two class in the client file named ASYrpc and DefRPC. These classes have invoke and get_result method. Asynchronous RPC separates a rpc from its return value. Whereas, deferred synchronous RPC is a process where client send the request to the server and waits only for acknowledgement of request received and it continuous with its process of computing the data.

TEAM CONTRIBUTION:

Worked on understanding the installation process of LINUX, file operation, synchronized storage, and synchronous, asynchronous and deferred RPC. Equal contribution in every part of the project in creating and demonstrating our skill in Distributed system for file operations and handling.

GAURAV CHAJJED MAHAVEER – 50%

SHYAM JIVANDAS RABADIYA – 50%

The whole project was done with equal communication and commitment between the team members on Microsoft Teams with valid proof of screenshots in every parts Outputs.

Spent 5 days by contributing 3 hrs each day for the completion of the Project 1.

Communication Mode: Microsoft Teams and UTA Library

REFERENCES:

- https://www.youtube.com/watch?v=3QiPPX-KeSc
- Python Socket Programming: Upload files from Client to Server - Stack Overflow
- https://cfrgtkky.blogspot.com/2019/01/python-socket-programming-upload-files.html
- https://github.com/munikarmanish/cse5306-project-1
- https://www.bing.com/videos/search?q=ubuntu&&view=detail&mid=A9D1BB0F4E861EE4FA11A9D1BB0F4E861EE4FA11&&FORM=VRDGAR&ru=%2Fvideos%2Fsearch%3Fq%3Dubuntu%26vsbkey%3Dubuntu%26vsbl%3D0%26FORM%3DSVRSRS
- https://techpiezo.com/linux/set-path-environment-variable-in-ubuntu/
- https://pymodbus.readthedocs.io/en/latest/source/example/synchronous_client.html
- https://thepythoncorner.com/posts/2019-01-13-how-to-create-a-watchdog-in-python-to-look-for-filesystem-changes/
- https://pythonhosted.org/watchdog/quickstart.html#quickstart
- https://stackoverflow.com/questions/38412887/how-to-send-a-list-through-tcp-sockets-python
- https://discuss.codecademy.com/t/can-we-add-a-new-column-at-a-specific-position-in-a-pandas-dataframe/355842
- https://www.kite.com/python/answers/how-to-get-the-indices-of-rows-in-a-pandas-dataframe-which-satisfy-a-given-condition-in-python
- https://stackoverflow.com/questions/40755680/how-to-reset-index-pandas-dataframe-after-dropna-pandas-dataframe
- https://stackoverflow.com/questions/13842088/set-value-for-particular-cell-in-pandas-dataframe-using-index
- https://www.marsja.se/how-to-get-the-column-names-from-a-pandas-dataframe-print-and-list/
- https://stackoverflow.com/questions/8384737/extract-file-name-from-path-no-matter-what-the-os-path-format
- https://help.resilio.com/hc/en-us/articles/205458165-Ignoring-files-in-Sync-Ignore-List-
- https://stackoverflow.com/questions/19071512/socket-error-errno-48-address-already-in-use