

11th session

- Modifying json file address
- Delete folder of the client who was deleted by admin
- Deleting a client username, password and privilege from json files when the client was deleted by admin
- Controlling “cd ..” to *only* be available if the user is not currently standing in Root
- Limiting each client to work in a directory of their own (a user can just enter to his/her directory and make changes)

The first problem one json files because we made it in root/server and when CWD get changed in a function (for example in change folder directory), we could not read or write json file there. And we have an error “the json file is not in this address”. So we try to save the CWD at the first of service module and we named it `init_cwd = str(os.getcwd())` and its equal the place of your project in your own computer, so when we use our client-server application in different computers, we would not face any problem. Then we use “`init_cwd/root/server/registered.json`” in address part when we want to read or write in json file.

Deleted client: while we were testing our application, we found that the deleted client could not registered again. Because we did not delete it in our json file and also we did not delete the directory of its own. So we modify our delete function and complete it to solve this problem. Fortunately, it works correctly and deleted client could register again, because they are unknown for server.

Controlling users when they are moving through folders was the last part we did in this session. We have to control they walk back to previous folder until root. So we put restrictions in going back function, we asked the current folder, if the client was in root we return an error “Error: You are in root directory”.

Next restriction was where they want to enter a folder, user can just enter user folder and then just his/her own directory, so we set a restriction, asked for the privilege and current place:

- 1) if it was user and stood in root, he/she just can enter user folder. If it was admin, he/she can go anywhere.
- 2) If it was user and stood in user folder, he/she just can enter his/her own folder (which name is the same as user name). If it was admin, he/she can go anywhere.
- 3) If it was not 2 last, it means the user is in his/her own directory and he can go anywhere he want. And he can do anything(changing folder, making folder, ...). If the folder name was not exist, the exception occur so we handle it with try exception, and return and error “Error: The folder does not exist.”