ONLINE QUIZ SYSTEM

Online Quiz System (using TCP Multithreading to cater unlimited clients at the same time) which contains three types of tests

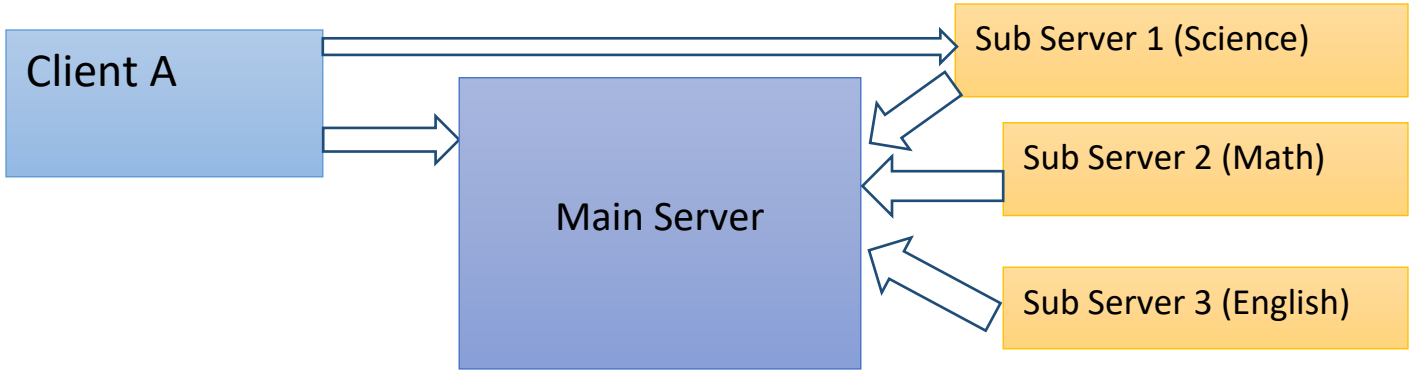
- Science.
- Mathematics.
- English.

Every test type further contains different areas as following

- Science → (Physics, Chemistry, Biology)
- Mathematics → (Geometry, Algebra, IQ)
- English → (Analogies, Antonyms, RC Questions)

Details:

There will be a main server and three sub-servers (for three different tests). The main server does not want to overburden itself by communicating with all the incoming clients and handle the tests, so it only stores the info about which sub-server contains what sort of test (Three sub-servers each of which contains one type of test i.e., Science, Math or English). When a sub-server connects to the main server it sends the main server a string containing the Test Name which it can carry out along with its IP and Port number (e.g. Mathematics, Sub-server IP, Sub-server Port). Whenever a client connects to the main server, it will display three kinds of tests (Science, Mathematics and English) to the client. The client will then choose the type of test (e.g., Math). The main server will then send the information of corresponding sub-server to the client. The client will then connect to the corresponding sub-server. The sub-server will then show different types of the corresponding test (e.g, Geometry, Algebra and IQ for Math Sub-server) to the client. The client will select the test type and complete the test (You don't need any test. Just make the sub-server to assign random marks to the client from 1 to 100). Then sub-server will send the information related to client (Client IP, Client Port, Test Name, Test Type, Marks) back to main server and also to the client. The client will then terminate. The main server will store the information within a file for all the clients.



1. Design a Main Server.
2. Design a Client.
3. Design 3 TCP Sub-servers.

The string a sub-server send to the main server will have the following format:

Test_name, Sub-server IP, Sub-server Port

4. Each sub-server will connect to main server. Main server will make a database about the sub-server information it gets from the sub-servers.
5. When client connects with main server and makes a request to give a specific test (e.g., Math), main server will search the record in the database it has for the corresponding sub-server and will send sub-server information to the client. The string a main server will send to the client will have the following format:

   Test_name, Sub-server IP, Sub-server Port

6. Using that information the client will reconnect to the sub server using TCP connection (Note: more than one clients can be connected to any particular sub server at a single time). The sub-server will then show different test-types to client. Client will select one and sub-server will assign random marks to client on that test type in the range of 1 to 100.

7. The sub server will send the final result to the client and also to the main server for the record of that client. The format of the information will be as follows:

   Client IP, Client Port, Test Name, Test Type, Marks

8. The client will show the result on the terminal and then quit. The main server will store the information in a file with the below mentioned format and keep listening for next clients.

   Client IP, Client Port, Test Name, Test Type, Marks

NOTE:

This was just a semester project so there are not any questions included and marks assigned at the random. This can be extended to add question of your desire and mark them according to your criteria.