

SOCKET PROGRAMMING

VIRTUAL COVID ASSISTANT

ABSTRACT

TEAM MEMBERS:

No	Name	SRN
1.	ADITYA S RAJ	PES1UG20CS622
2.	LIKITH R	PES1UG20CS659
3.	TOUKHEER BAIG	PES1UG20CS616

DESCRIPTION:

With large scale global pandemics and other unruly human diseases coming into picture, it is statistically impossible to have a check on every suspecting patient with human effort. Thereby, a simple end to end host communication between an automated assisted guide and the suspecting patient is made possible from virtually anywhere.

A medical research induced data questionnaire is the base of this project implementation.

The assistant is expected to always give a perfect and quick diagnosis for the patient. This can be achieved by using a decision tree for its proven efficiency. A pre-known set of questions and its right predicted answers in binary pickle-d form ensures the accuracy and thus falls under the class of supervised learning. Based on the symptoms that the patient is showing, they would pick what suits the situation best. After obtaining the said information, the server processes the data to give the best possible diagnosis.

To achieve this as a structure of end user hosts and networks, the Transmission Control Protocol works best for the requirement. Being almost perfectly reliable, it makes sure that there is no failure in the transfer of data between the client(patient) and the server(assistant). A failure in the transfer of data packets or any loss that's endured may lead to a totally different diagnosis in this sensitive problem of human healthcare.

Any other protocol like UDP follows best effort delivery mechanism and can prove to be of no use in case there's any loss.

Also, to enable multiple patients to get assistance at the very same time, the concept of threads is used. Each client(patient) request to connect to the server(assistant) is treated as a new thread and resources are allocated independently. Resources for a particular client cannot be reserved forever when it is not being used as this would lead to a resource imbalance. That is why an automatic timer makes sure that any unresponsive client is disconnected from the server, freeing said resources.

Finally, a simple but understandable Graphical user interface is automatically developed using the PyQt5 software. This is established for the general eye catching look of the user window while the basic functionality remains unchanged.

IMPORTANT MODULES AND PACKAGES:

- sklearn for the decision tree used
- pickle to encode the binary data
- numpy to scale the encoded data
- PyQt5 for the graphical user interface
- response for the client-server request response mechanism
- threading for multiple client connection
- socket to build the actual TCP socket
- time to track any unresponsive clients
- errno to define erroneous conditions

RESOURCES:

<https://stackoverflow.com/questions/2719017/how-to-set-timeout-on-pythons-socket-recv-method>

<http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>

<https://realpython.com/python-sockets/>

<https://www.v7labs.com/blog/train-validation-test-set>