**Project** :- Create program to send data to a digital board containing Rasberry pi

Day 1:

Understand socket programming

Day 2:

Understand the python code for creating server and client

Link to reading material-

<https://www.geeksforgeeks.org/socket-programming-python/>

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

Day 3:

Send a string to the local host using the client and server created

Day 4:

Send a file using ftp server

Reading material –

For creating server and client

<https://www.bogotobogo.com/python/python_network_programming_server_client_file_transfer.php>

file reading - <https://www.w3schools.com/python/ref_func_open.asp>

Day 5(7/9/19):

Unsuccessful in sending the image using ftp server.

Got encoding error.

Need to encode in a proper format for the image.

Hence using json to do the job.

Hope it works.

Json important functions :

Json.dumps() -- The json.dumps() function (which means “dump string,” not “dumps”) will

translate a Python value into a string of JSON-formatted data.

Json.loads() --To translate a string containing JSON data into a Python value, pass it to

the json.loads() function. (The name means “load string,” not “loads.”)

**To convert .ui into .py run this command in cmd: -**

Pyuic5 –x filename.ui -o outputFileName.py

**Successfully tested the client and server backend code on raspberry pi on 7/10/19**

Part 2 of Project :- Making of the GUI

For making the GUI following link was used :

<http://www.science.smith.edu/dftwiki/index.php/PyQt5_Tutorial:_A_Window_Application_with_File_IO>

Steps –

1. Uses Designer app shipped with PyQt5 to create the GUI using drag and drop mechanism.
2. Connect the button, texts and labels in the GUI to the slots.
3. Designer will create a .ui file, convert it using the above command to a .py file.
4. Edit the .py file so that it contains functions for the slots.
5. Use Model-View-Presenter architecture to connect the GUI with the backend.