EXPERIMENT 8

NAME: Chirag Rana **UID: 2018130043**
CLASS: TE COMPS **BATCH: C**
DATE: 6/11/2020

Aim: To implement Socket Programming and establish a connection between Client and server.

Theory:
Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server.

State diagram for server and client model

Client Code in Java using Android:

```
package com.example.alarmswitch;

import android.net.UrlQuerySanitizer;
import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.text.format.Formatter;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import okhttp3.Call;
import okhttp3.Callback;
import okhttp3.Request;
import okhttp3.RequestBody;
import okhttp3.Response;
import androidx.appcompat.app.AppCompatActivity;


import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.util.HashMap;
import java.util.Map;

import okhttp3.MediaType;
import okhttp3.OkHttpClient;
```

```
import okhttp3.RequestBody;

public class MainActivity extends AppCompatActivity {

    private Boolean ALARM_STATUS=true;


    private String url = "http://" + "192.168.43.241" + ":" + 5000 + "/";

    private String postBodyString;
    private MediaType mediaType;
    private RequestBody requestBody;
    private Button alarm;
    private Integer status = 0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        alarm = findViewById(R.id.alarm_control);
        alarm.setText("STOP");

        alarm.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                String username = getIntent().getStringExtra("username");
                String password=getIntent().getStringExtra("password");

                try {
                    postRequest("{status:"+status.toString()+",username:"+username+",password:
"+password+"}", url);
                    status = (status + 1)%2;
                    if (alarm.getText() == "STOP"){
                        alarm.setText("START");
                    }
                    else{
                        alarm.setText("STOP");
                    }


                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
```

```
        });

    }
    private RequestBody buildRequestBody(JSONObject msg) {
        JSONObject json = msg;
        mediaType = MediaType.parse("application/json; charset=utf-8");
        requestBody = RequestBody.create(String.valueOf(json), mediaType);
        return requestBody;
    }

    private void postRequest(String postBody, String postUrl ) throws JSONException {

        OkHttpClient client = new OkHttpClient();
        JSONObject obj = new JSONObject(postBody);
        RequestBody requestBody = buildRequestBody(obj);

        Log.d("TAG", postBody);
        Request request = new Request.Builder()
                .post(requestBody)
                .url(postUrl)
                .build();

        client.newCall(request).enqueue(new Callback() {
            @Override
            public void onFailure(final Call call, final IOException e) {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {

                        Log.d("TAG", url);
                        Toast.makeText(MainActivity.this, "Something went wrong:" + " " +
e.getMessage(), Toast.LENGTH_LONG).show();
                        Log.d("TAG","Something went wrong:" + " " + e.getMessage());


                        call.cancel();


                    }
                });

            }

            @Override
            public void onResponse(Call call, final Response response) throws IOException {
```

```
                runOnUiThread(new Runnable() {
                    @Override

                    public void run() {
                        if (status == 1){
                            Toast.makeText(MainActivity.this, "You have Turned Off the Alarm.",
Toast.LENGTH_SHORT).show();
                        }else{
                            Toast.makeText(MainActivity.this, "You have Turned On the Alarm.",
Toast.LENGTH_SHORT).show();

                        }
                        }
                    }
                );


            }
        });
    }




}
```

```
D/NetworkSecurityConfig: No Network Security Config specified, using platform default
D/TAG: {status:0,username:Chirag,password: Rana}
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/TAG: {status:1,username:Chirag,password: Rana}
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/TAG: {status:0,username:Chirag,password: Rana}
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/TAG: {status:1,username:Chirag,password: Rana}
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
D/EGL_emulation: eglMakeCurrent: 0xddd85300: ver 2 0 (tinfo 0xddd83650)
```

SERVER SIDE Code:

```
from flask import Flask, request, jsonify
```

```
import os

app = Flask(__name__)
basedir = os.path.abspath(os.path.dirname(__file__))
stat = False

@app.route('/', methods=['POST'])
def post_data():
    print(request.get_json())
    return jsonify({'status':'200'})

@app.route('/', methods=['GET'])
def get_data():

    return {'status':'200'}


if __name__ == '__main__':
    app.run(host="0.0.0.0", port=5000, debug=True, use_reloader=False)
```

```
2020-11-06 17:10:08.697148: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1263]
New height: 384 new width: 644
{'status': 0, 'username': 'Chirag', 'password': 'Rana'}
127.0.0.1 - - [06/Nov/2020 17:10:13] "POST / HTTP/1.1" 200 -
{'status': 1, 'username': 'Chirag', 'password': 'Rana'}
127.0.0.1 - - [06/Nov/2020 17:10:16] "POST / HTTP/1.1" 200 -
{'status': 0, 'username': 'Chirag', 'password': 'Rana'}
127.0.0.1 - - [06/Nov/2020 17:10:16] "POST / HTTP/1.1" 200 -
```

Conclusion: Implemented a client and server connection using Android as client and flask as the server.