**Hardware code**

#include <Wire.h>

#include <LiquidCrystal.h>

const int RS = 2;

const int E = 3;

const int D4 = 4;

const int D5 = 5;

const int D6 = 6;

const int D7 = 7;

LiquidCrystal lcd(RS, E, D4, D5, D6, D7);

int buzzer = 12;

const int analogtemp = A0;

double temp=0, Vin=0, samples[250];

unsigned int j=0;

const double Vref=1100.0;

int pulsePin = A2;

int blinkPin = 13;

int fadePin = 8;

int fadeRate = 0;

volatile int BPM;

volatile int spo;

volatile int Signal;

volatile int IBI = 600;

volatile boolean Pulse = false;

volatile boolean QS = false;

static boolean serialVisual = true;

volatile int rate[10];

volatile unsigned long sampleCounter = 0;

volatile unsigned long lastBeatTime = 0;

volatile int P = 512;

volatile int T = 512;

volatile int thresh = 525;

volatile int amp = 100;

volatile boolean firstBeat = true;

volatile boolean secondBeat = false;

void setup() {

analogReference(INTERNAL);

pinMode(buzzer, OUTPUT);

pinMode(blinkPin,OUTPUT);

pinMode(fadePin,OUTPUT);

interruptSetup();

digitalWrite(buzzer,HIGH);

lcd.begin(16, 2);

lcd.setCursor(0,0);

lcd.print(" SMART HEALTH ");

lcd.setCursor(0,1);

lcd.print(" MONITORING ");

delay(1000);

Serial.begin(9600);

delay(1000);

digitalWrite(buzzer,LOW);

lcd.clear();

}

void loop() {

Vin=0;

temp=0;

for(j=0 ; j<=249 ; j++)

{

samples[j] = (analogRead(analogtemp));

Vin = Vin + ( samples[j] \* Vref/1024.0);

}

Vin=Vin/250.0;

int tempvv = (Vin/10.0);

lcd.setCursor(0,0);

lcd.print("T:");

lcd.print(tempvv);

lcd.print(" ");

lcd.setCursor(6,0);

lcd.print("HR:");

lcd.print(BPM);

lcd.print(" ");

lcd.setCursor(0,1);

lcd.print("O:");

lcd.print(spo);

lcd.print(" ");

if(tempvv > 39)

{

digitalWrite(buzzer,HIGH);

}

else

{

digitalWrite(buzzer,LOW);

}

Serial.print(tempvv);

Serial.print(" ");

Serial.print(BPM);

Serial.print(" ");

Serial.print(spo);

Serial.print(" ");

Serial.println();

delay(1000);

serialOutput();

if (QS == true)

{

fadeRate = 255;

serialOutputWhenBeatHappens();

QS = false;

}

ledFadeToBeat();

delay(20);

}

void ledFadeToBeat()

{

fadeRate -= 15;

fadeRate = constrain(fadeRate,0,255); analogWrite(fadePin,fadeRate);

}

void interruptSetup()

{

TCCR2A = 0x02;

TCCR2B = 0x06;

OCR2A = 0X7C;

TIMSK2 = 0x02;

sei();

}

void serialOutput()

{

if (serialVisual == true)

{

arduinoSerialMonitorVisual('-', Signal); }

else

{

sendDataToSerial('S', Signal);

}

}

void serialOutputWhenBeatHappens()

{

if (serialVisual == true) {

Serial.print("\*\*\* Heart-Beat Happened \*\*\* "); Serial.print("BPM: ");

Serial.println(BPM);

lcd.clear();

}

else

{

sendDataToSerial('B',BPM);

sendDataToSerial('Q',IBI);

}

}

ISR(TIMER2\_COMPA\_vect)

{

cli();

Signal = analogRead(pulsePin);

sampleCounter += 2;

int N = sampleCounter - lastBeatTime;

if(Signal < thresh && N > (IBI/5)\*3) {

if (Signal < T)

{

T = Signal;

}

}

if(Signal > thresh && Signal > P)

{

P = Signal;

}

if (N > 250)

{

if ( (Signal > thresh) && (Pulse == false) && (N > (IBI/5)\*3) )

{

Pulse = true; digitalWrite(blinkPin,HIGH);

IBI = sampleCounter - lastBeatTime; lastBeatTime = sampleCounter;

if(secondBeat)

{

secondBeat = false;

for(int i=0; i<=9; i++) {

rate[i] = IBI;

}

}

if(firstBeat)

{

firstBeat = false;

secondBeat = true;

sei();

return;

}

word runningTotal = 0;

for(int i=0; i<=8; i++)

{

rate[i] = rate[i+1];

runningTotal += rate[i];

}

rate[9] = IBI;

runningTotal += rate[9];

runningTotal /= 10;

BPM = 60000/runningTotal;

QS = true;

}

}

if (Signal < thresh && Pulse == true)

{

digitalWrite(blinkPin,LOW);

Pulse = false;

amp = P - T;

thresh = amp/2 + T;

P = thresh;

T = thresh;

}

if (N > 2500)

{

thresh = 512;

P = 512;

T = 512;

lastBeatTime = sampleCounter;

firstBeat = true;

secondBeat = false;

}

sei();

}

**Wifi code**

#include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>

#include <String.h>

#include <TextFinder.h>

TextFinder finder(Serial);

const int no\_of\_fields = 3;

int fieldID = 0;

int values[no\_of\_fields];

const char\* ssid = "yourid";

const char\* pass = "password";

String ipaddress = "http://192.168.1.17/smarthealth/update.php?";

void setup () {

delay(2000);

pinMode(LED\_BUILTIN, OUTPUT);

Serial.begin(115200);

digitalWrite(LED\_BUILTIN, HIGH);

WiFi.begin(ssid, pass);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.print("Connecting...");

}

digitalWrite(LED\_BUILTIN, LOW);

}

void loop() {

if (WiFi.status() == WL\_CONNECTED) {

for(fieldID = 0; fieldID < 3; fieldID ++)

{

values[fieldID] = finder.getValue();

}

int data1 = values[0];

int data2 = values[1];

int data3 = values[2];

digitalWrite(LED\_BUILTIN, HIGH);

String postStr = ipaddress;

postStr +="data1=";

postStr +=String(data1);

postStr +="&data2=";

postStr +=String(data2);

postStr +="&data3=";

postStr +=String(data3);

fieldID = 0;

HTTPClient http;

HTTPClient

Serial.println(postStr);

http.begin(postStr);

int httpCode = http.GET();

if (httpCode > 0) {

String payload = http.getString();

Serial.println(payload);

}

http.end();

}

digitalWrite(LED\_BUILTIN, LOW);

delay(1000);

}

**Software code**

UserHomeActivity.java

package com.example.smarthealthmonitoringmat;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import androidx.fragment.app.Fragment;

import android.app.AlertDialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.view.MenuItem;

import com.example.smarthealthmonitoringmat.history.HistoryFragment;

import com.example.smarthealthmonitoringmat.home.HomeFragment;

import com.google.android.material.bottomnavigation.BottomNavigationView;

public class UserHomeActivity extends AppCompatActivity implements BottomNavigationView.OnNavigationItemSelectedListener {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_user\_home);

loadFragment(new HomeFragment());

BottomNavigationView navigation = findViewById(R.id.navigation);

navigation.setOnNavigationItemSelectedListener(this);

}

@Override

public boolean onNavigationItemSelected(@NonNull MenuItem item) {

Fragment fragment = null;

switch (item.getItemId()) {

case R.id.navigation\_home:

fragment = new HomeFragment();

break;

case R.id.navigation\_notifications:

fragment = new HistoryFragment();

break;

case R.id.navigation\_logout:

showPopup();

break;

}

return loadFragment(fragment);

}

private boolean loadFragment(Fragment fragment) {

//switching fragment

if (fragment != null) {

getSupportFragmentManager()

.beginTransaction()

.replace(R.id.fragment\_container, fragment)

.commit();

return true;

}

return false;

}

private void showPopup() {

AlertDialog.Builder alert = new AlertDialog.Builder(UserHomeActivity.this);

alert.setMessage("Are you sure?")

.setPositiveButton("Logout", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

logout(); // Last step. Logout function

}

}).setNegativeButton("Cancel", null);

AlertDialog alert1 = alert.create();

alert1.show();

}

private void logout() {

Intent intentMain=new Intent(getApplicationContext(), LoginActivity.class);

intentMain.addFlags(intentMain.FLAG\_ACTIVITY\_NEW\_TASK | intentMain.FLAG\_ACTIVITY\_CLEAR\_TASK);

startActivity(intentMain);

finish();

}

}

**RegisterActivity.java**

package com.example.smarthealthmonitoringmat;

import android.app.DatePickerDialog;

import android.app.ProgressDialog;

import android.content.Intent;

import android.os.AsyncTask;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.DatePicker;

import android.widget.EditText;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.io.DataInputStream;

import java.net.HttpURLConnection;

import java.net.URL;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Locale;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import es.dmoral.toasty.Toasty;

public class RegisterActivity extends AppCompatActivity {

EditText register\_name,register\_age,register\_dob,register\_contact,register\_email,register\_address,register\_username,register\_password,register\_cpassword;

TextView register\_btn,login\_btn;

String name,age,dob,contact,address,email,username,password,cpassword;

final Calendar myCalendar = Calendar.getInstance();

ProgressDialog pDialog;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_register);

register\_name=findViewById(R.id.user\_register\_name);

register\_dob=findViewById(R.id.user\_register\_dob);

register\_contact=findViewById(R.id.user\_register\_contact);

register\_email=findViewById(R.id.user\_register\_email);

register\_address=findViewById(R.id.user\_register\_address);

register\_username=findViewById(R.id.user\_register\_username);

register\_password=findViewById(R.id.user\_register\_password);

register\_cpassword=findViewById(R.id.user\_register\_cpassword);

register\_btn=findViewById(R.id.user\_register\_btn);

login\_btn=findViewById(R.id.login\_link);

DatePickerDialog.OnDateSetListener date=new DatePickerDialog.OnDateSetListener() {

@Override

public void onDateSet(DatePicker datePicker, int year, int month, int day) {

myCalendar.set(Calendar.YEAR,year);

myCalendar.set(Calendar.MONTH,month);

myCalendar.set(Calendar.DAY\_OF\_MONTH,day);

updateLabel();

}

};

register\_dob.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

new DatePickerDialog(RegisterActivity.this, date, myCalendar

.get(Calendar.YEAR), myCalendar.get(Calendar.MONTH),

myCalendar.get(Calendar.DAY\_OF\_MONTH)).show();

}

});

register\_btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

validation();

}

});

login\_btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), MainActivity.class));

}

});

}

private void updateLabel() {

String myFormat = "yyyy-MM-dd"; //In which you need put here

SimpleDateFormat sdf = new SimpleDateFormat(myFormat, Locale.US);

register\_dob.setText(sdf.format(myCalendar.getTime()));

}

private void validation()

{

name=register\_name.getText().toString();

dob=register\_dob.getText().toString();

contact=register\_contact.getText().toString();

email=register\_email.getText().toString();

address=register\_address.getText().toString();

username=register\_username.getText().toString();

password=register\_password.getText().toString();

cpassword=register\_cpassword.getText().toString();

if (name.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter name!",Toasty.LENGTH\_SHORT).show();

}

else if (dob.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Date-of-Birth!",Toasty.LENGTH\_SHORT).show();

}

else if (contact.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Contact!",Toasty.LENGTH\_SHORT).show();

}

else if (email.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Email!",Toasty.LENGTH\_SHORT).show();

}

else if (address.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Address!",Toasty.LENGTH\_SHORT).show();

}

else if (username.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Username!",Toasty.LENGTH\_SHORT).show();

}

else if (password.isEmpty())

{

Toasty.warning(getApplicationContext(),"Enter Password!",Toasty.LENGTH\_SHORT).show();

}

else if (password.equalsIgnoreCase(cpassword))

{

boolean s = checkEmail(register\_email.getText().toString());

boolean c=isValidPhoneNumber(register\_contact.getText().toString());

boolean p=isValidPassword(register\_password.getText().toString());

if (c)

{

if (s)

{

if (p)

{

new reg().execute();

}

else

{

Toasty.warning(getApplicationContext(),"password Credential! ex(user@123)",Toasty.LENGTH\_SHORT).show();

}

}

else

{

Toasty.warning(getApplicationContext(),"Invalid Email!",Toasty.LENGTH\_SHORT).show();

}

}

else

{

Toasty.warning(getApplicationContext(),"Invalid contact!",Toasty.LENGTH\_SHORT).show();

}

}

else

{

Toasty.warning(getApplicationContext(),"Password mismatch!",Toasty.LENGTH\_SHORT).show();

}

}

public static boolean checkEmail(String eml) {

Pattern EMAIL\_ADDRESS\_PATTERN = Pattern

.compile("[a-zA-Z0-9+.\_%-+]{1,256}" + "@"

+ "[a-zA-Z0-9][a-zA-Z0-9-]{0,64}" + "(" + "."

+ "[a-zA-Z0-9][a-zA-Z0-9-]{0,25}" + ")+");

return EMAIL\_ADDRESS\_PATTERN.matcher(eml).matches();

}

private boolean isValidPhoneNumber(String phone) {

Pattern p = Pattern.compile("(0/91)?[5-9][0-9]{9}");

Matcher m = p.matcher(phone);

return (m.find() && m.group().equals(phone));

}

public static boolean isValidPassword(String password)

{

// Regex to check valid password.

String regex = "^(?=.\*[0-9])"+ "(?=.\*[a-z])(?=.\*[A-Z])"+ "(?=.\*[@#$%^&+=])"+ "(?=\\S+$).{8,20}$";

Pattern p = Pattern.compile(regex);

Matcher m = p.matcher(password);

return m.matches();

}

public class reg extends AsyncTask<String, String, String>

{

String reg\_name=register\_name.getText().toString();

String reg\_dob=register\_dob.getText().toString();

String reg\_contact=register\_contact.getText().toString();

String reg\_email=register\_email.getText().toString();

String reg\_address=register\_address.getText().toString();

String reg\_password=register\_password.getText().toString();

String reg\_username=register\_username.getText().toString();

@Override

protected void onPreExecute() {

super.onPreExecute();

pDialog = new ProgressDialog(RegisterActivity.this);

pDialog.setMessage("Requesting " + reg\_name + ")...");

pDialog.setIndeterminate(false);

pDialog.setCancelable(false);

pDialog.show();

}

@Override

protected String doInBackground(String... strings) {

String txt = "";

try {

String ur = "http://"+ MainActivity.sip+"/register1.php?"+ "name=" + reg\_name

+ "&dob=" +reg\_dob+ "&contact=" +reg\_contact

+"&email="+reg\_email+"&address="+reg\_address+"&username="+reg\_username +"&password="+reg\_password ;

Log.i("URL", ur);

URL url = new URL(ur);

HttpURLConnection uc = (HttpURLConnection) url.openConnection();

DataInputStream dis = new DataInputStream(uc.getInputStream());

String t = "";

while ((t = dis.readLine()) != null) {

txt += t;

}

dis.close();

} catch (Exception e) {

Log.i("Login Ex", e.getMessage());

}

return txt;

}

@Override

protected void onPostExecute(String s) {

super.onPostExecute(s);

Log.i("file\_url", s);

String tmp=s;

if (s.trim().equals("success"))

{

Toasty.success(getApplicationContext(), "Register Successfully!", Toasty.LENGTH\_SHORT).show();

Intent in = new Intent(getApplicationContext(), MainActivity.class);

startActivity(in);

}

else if(s.trim().equals("Invalid user")) {

Toasty.warning(getApplicationContext(), "Invalid user", Toasty.LENGTH\_SHORT).show();

}

else

{ Toasty.warning(getApplicationContext(), "Please Check ...!", Toasty.LENGTH\_SHORT).show();}

pDialog.dismiss();

}

}

}

**MainActivity.java**

package com.example.smarthealthmonitoringmat;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.text.TextUtils;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

EditText ip\_address;

Button connect\_btn;

public static String sip="";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

ip\_address=findViewById(R.id.ipaddress);

connect\_btn=findViewById(R.id.ip\_btn);

ip\_address.setText("192.168.1.7");

connect\_btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

sip=ip\_address.getText().toString().trim();

if(TextUtils.isEmpty(sip))

{

Toast.makeText(MainActivity.this, "Enter Ip Address", Toast.LENGTH\_SHORT).show();

}

else

{

sip=ip\_address.getText().toString().trim()+"/smarthealth";

Intent intent=new Intent(getApplicationContext(),LoginActivity.class);

startActivity(intent);

}

}

});

}

}

**LoginActivity.java**

package com.example.smarthealthmonitoringmat;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.AsyncTask;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import java.io.DataInputStream;

import java.net.HttpURLConnection;

import java.net.URL;

public class LoginActivity extends AppCompatActivity {

TextView user\_login,register;

EditText user\_username,user\_password;

public static String gemail="";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_login);

user\_login=findViewById(R.id.user\_login\_btn);

register=findViewById(R.id.register\_limk);

user\_username=findViewById(R.id.user\_login\_username);

user\_password=findViewById(R.id.user\_login\_password);

user\_username.setText("admin");

user\_password.setText("11");

user\_login.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

validation();

}

});

register.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(),RegisterActivity.class));

}

});

}

private void validation()

{

String username=user\_username.getText().toString();

String password=user\_password.getText().toString();

if (username.isEmpty())

{

Toast.makeText(getApplicationContext(),"Enter Username!",Toast.LENGTH\_SHORT).show();

}

else if (password.isEmpty())

{

Toast.makeText(getApplicationContext(),"Enter password",Toast.LENGTH\_SHORT).show();

}

else

{

new userlogin().execute();

}

}

public class userlogin extends AsyncTask<String, String, String> {

String lname=user\_username.getText().toString();

String lpass=user\_password.getText().toString();

@Override

protected void onPreExecute() {

super.onPreExecute();

}

protected String doInBackground(String... args) {

String txt = "";

try {

String ur = "http://"+ MainActivity.sip+"/userlogin.php?"+ "uname=" + lname + "&pword=" +lpass;

URL url = new URL(ur);

Log.i("URL", ""+url);

HttpURLConnection uc = (HttpURLConnection) url.openConnection();

DataInputStream dis = new DataInputStream(uc.getInputStream());

String t = "";

while ((t = dis.readLine()) != null) {

txt += t;

}

Log.i("Read", txt);

// m=txt;

dis.close();

} catch (Exception e) {

Log.i("Login Ex", e.getMessage());

}

return txt;

}

protected void onPostExecute(String file\_url) {

Log.i("file\_url", file\_url);

if (file\_url.trim().equals("success")) {

gemail=lname;

Toast.makeText(getApplicationContext(), "Login Success", Toast.LENGTH\_SHORT).show();

Intent in = new Intent(getApplicationContext(), UserHomeActivity.class);

startActivity(in);

}

else if(file\_url.trim().equals("Invalid user")) {

Toast.makeText(getApplicationContext(), "Invalid user", Toast.LENGTH\_SHORT).show();

}

else

{

Toast.makeText(getApplicationContext(), "Please Check Login...!", Toast.LENGTH\_SHORT).show();

}

}

}

}

**HomeFragment.java**

package com.example.smarthealthmonitoringmat.home;

import android.app.Activity;

import android.app.ProgressDialog;

import android.os.AsyncTask;

import android.os.Bundle;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.fragment.app.Fragment;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ListAdapter;

import android.widget.ListView;

import android.widget.TextView;

import com.example.smarthealthmonitoringmat.MainActivity;

import com.example.smarthealthmonitoringmat.R;

import com.example.smarthealthmonitoringmat.UserHomeActivity;

import org.apache.http.HttpEntity;

import org.apache.http.HttpResponse;

import org.apache.http.client.methods.HttpPost;

import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.http.params.BasicHttpParams;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.io.BufferedReader;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.HashMap;

public class HomeFragment extends Fragment {

UserHomeActivity activity;

ProgressDialog pDialog;

JSONArray transactions = null;

ArrayList<HashMap<String, String>> transactionlist;

ListView list;

String userid;

ListAdapter adapter;

String myJSON;

private static final String TAG\_RESULTS = "result";

private static final String TAG\_DATA1 = "data1";

private static final String TAG\_DATA2 = "data2";

private static final String TAG\_DATA3 = "data3";

private static final String TAG\_DATA4 = "data4";

private static final String TAG\_DATA5 = "data5";

private static final String TAG\_DATA6 = "data6";

TextView sensor1,sensor2,sensor3,sensor4,sensor5,sensor6;

String data1,data2,data3,data4,data5,data6;

Thread thread;

private volatile boolean exit = false;

@Override

public void onAttach(@NonNull Activity activity) {

super.onAttach(activity);

this.activity=(UserHomeActivity)activity;

}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

// Inflate the layout for this fragment

View root= inflater.inflate(R.layout.fragment\_home, container, false);

sensor1=root.findViewById(R.id.sensor1);

sensor2=root.findViewById(R.id.sensor2);

sensor3=root.findViewById(R.id.sensor3);

transactionlist = new ArrayList<HashMap<String, String>>();

thread = new Thread() {

@Override

public void run() {

try {

while (!thread.isInterrupted()) {

Thread.sleep(3000);

// thread.stop();

activity.runOnUiThread(new Runnable() {

@Override

public void run() {

// update TextView here!

getNews();

//Toast.makeText(SensorActivity.this, "reloaded", Toast.LENGTH\_SHORT).show();

}

});

}

} catch (InterruptedException e) {

}

}

};

thread.start();

return root;

}

private void getNews() {

class GetDataJSON extends AsyncTask<String, Void, String> {

@Override

protected String doInBackground(String... params) {

DefaultHttpClient httpclient = new DefaultHttpClient(new BasicHttpParams());

HttpPost httppost = new HttpPost("http://" + MainActivity.sip + "/view1.php");

Log.i("httppost", "http://" + MainActivity.sip + "/view1.php");

// Depends on your web service

httppost.setHeader("Content-type", "application/json");

InputStream inputStream = null;

String result = null;

try {

HttpResponse response = httpclient.execute(httppost);

HttpEntity entity = response.getEntity();

inputStream = entity.getContent();

// json is UTF-8 by default

BufferedReader reader = new BufferedReader(new InputStreamReader(inputStream, "UTF-8"), 8);

StringBuilder sb = new StringBuilder();

String line = null;

while ((line = reader.readLine()) != null) {

sb.append(line + "\n");

}

result = sb.toString();

Log.i("read", result);

} catch (Exception e) {

// Oops

} finally {

try {

if (inputStream != null) inputStream.close();

} catch (Exception squish) {

}

}

return result;

}

@Override

protected void onPostExecute(String result) {

myJSON = result;

// showList();

try {

JSONObject jsonObj = new JSONObject(myJSON);

transactions = jsonObj.getJSONArray(TAG\_RESULTS);

int lo = transactions.length();

System.out.print(lo);

for (int i = 0; i < transactions.length(); i++) {

JSONObject c = transactions.getJSONObject(i);

data1 = c.getString(TAG\_DATA1);

data2 = c.getString(TAG\_DATA2);

data3 = c.getString(TAG\_DATA3);

Log.i("data1", data1);

Log.i("data2", data2);

Log.i("data3", data3);

HashMap<String, String> persons = new HashMap<String, String>();

persons.put(TAG\_DATA1, data1);

persons.put(TAG\_DATA2, data2);

persons.put(TAG\_DATA3, data3);

transactionlist.add(persons);

sensor1.setText(data1);

sensor2.setText(data2);

sensor3.setText(data3);

}

} catch (JSONException e) {

e.printStackTrace();

}

}

}

GetDataJSON g = new GetDataJSON();

g.execute();

}

}

**HistoryFragment.java**

package com.example.smarthealthmonitoringmat.history;

import android.app.Activity;

import android.content.Intent;

import android.os.AsyncTask;

import android.os.Bundle;

import androidx.annotation.NonNull;

import androidx.fragment.app.Fragment;

import android.os.StrictMode;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.Button;

import android.widget.ListAdapter;

import android.widget.ListView;

import android.widget.SimpleAdapter;

import android.widget.TextView;

import android.widget.Toast;

import com.example.smarthealthmonitoringmat.LoginActivity;

import com.example.smarthealthmonitoringmat.MainActivity;

import com.example.smarthealthmonitoringmat.R;

import com.example.smarthealthmonitoringmat.UserHomeActivity;

import org.apache.http.HttpEntity;

import org.apache.http.HttpResponse;

import org.apache.http.client.methods.HttpPost;

import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.http.params.BasicHttpParams;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.io.BufferedReader;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.HashMap;

public class HistoryFragment extends Fragment {

UserHomeActivity activity;

String myJSON;

private static final String TAG\_RESULTS = "result";

private static final String TAG\_ID = "id";

private static final String TAG\_DATE = "date";

private static final String TAG\_TIME = "time";

private static final String TAG\_DATA1 = "data1";

private static final String TAG\_DATA2 = "data2";

private static final String TAG\_DATA3 = "data3";

JSONArray transactions = null;

ArrayList<HashMap<String, String>> transactionlist;

ListView list;

String userid;

ListAdapter adapter;

String id,date,time,data1,data2,data3;

@Override

public void onAttach(@NonNull Activity activity) {

super.onAttach(activity);

this.activity=(UserHomeActivity)activity;

}

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container,

Bundle savedInstanceState) {

// Inflate the layout for this fragment

View root= inflater.inflate(R.layout.fragment\_history, container, false);

if(android.os.Build.VERSION.SDK\_INT > 9)

{

StrictMode.ThreadPolicy policy = new

StrictMode.ThreadPolicy.Builder().permitAll().build();

StrictMode.setThreadPolicy(policy);

}

userid = LoginActivity.gemail;

list =root.findViewById(R.id.list);

transactionlist = new ArrayList<HashMap<String, String>>();

getNews();

return root;

}

private void getNews() {

class GetDataJSON extends AsyncTask<String, Void, String> {

@Override

protected String doInBackground(String... params) {

DefaultHttpClient httpclient = new DefaultHttpClient(new BasicHttpParams());

Log.i("accno", userid);

HttpPost httppost = new HttpPost("http://" + MainActivity.sip + "/view\_history.php");

Log.i("httppost", "http://" + MainActivity.sip + "/view\_history.php");

// Depends on your web service

httppost.setHeader("Content-type", "application/json");

InputStream inputStream = null;

String result = null;

try {

HttpResponse response = httpclient.execute(httppost);

HttpEntity entity = response.getEntity();

inputStream = entity.getContent();

// json is UTF-8 by default

BufferedReader reader = new BufferedReader(new InputStreamReader(inputStream, "UTF-8"), 8);

StringBuilder sb = new StringBuilder();

String line = null;

while ((line = reader.readLine()) != null) {

sb.append(line + "\n");

}

result = sb.toString();

Log.i("read", result);

} catch (Exception e) {

// Oops

} finally {

try {

if (inputStream != null) inputStream.close();

} catch (Exception squish) {

}

}

return result;

}

@Override

protected void onPostExecute(String result) {

myJSON = result;

try {

JSONObject jsonObj = new JSONObject(myJSON);

transactions = jsonObj.getJSONArray(TAG\_RESULTS);

int lo = transactions.length();

System.out.print(lo);

for (int i = 0; i < transactions.length(); i++) {

JSONObject c = transactions.getJSONObject(i);

date = c.getString(TAG\_DATE);

id = c.getString(TAG\_ID);

time = c.getString(TAG\_TIME);

data1 = c.getString(TAG\_DATA1);

data2 = c.getString(TAG\_DATA2);

data3 = c.getString(TAG\_DATA3);

Log.i("data1", data1);

Log.i("data2", data2);

Log.i("data3", data3);

Log.i("date", date);

Log.i("hostid", id);

Log.i("time", time);

HashMap<String, String> persons = new HashMap<String, String>();

persons.put(TAG\_ID, id);

persons.put(TAG\_DATE, date);

persons.put(TAG\_TIME, time);

persons.put(TAG\_DATA1, data1);

persons.put(TAG\_DATA2, data2);

persons.put(TAG\_DATA3, data3);

adapter = new SimpleAdapter(

activity, transactionlist, R.layout.user\_list\_chk, new String[]{TAG\_DATE,TAG\_TIME,TAG\_DATA1,TAG\_DATA2,TAG\_DATA3,TAG\_ID}, new int[]{R.id.date,R.id.time,R.id.data1,R.id.data2,R.id.data3,R.id.id}) {

@Override

public View getView(final int position, View convertView, ViewGroup parent) {

View v = super.getView(position, convertView, parent);

{

}

return v;

}

};

list.setAdapter(adapter);

transactionlist.add(persons);

}

} catch (JSONException e) {

e.printStackTrace();

}

}

}

GetDataJSON g = new GetDataJSON();

g.execute();

}

}