#include <ESP8266WiFi.h>

#include <Wire.h>

#include <MCP3008.h>

#include "MAX30100\_PulseOximeter.h"

#define REPORTING\_PERIOD\_MS 1000

PulseOximeter pox;

float BPM, SpO2;

uint32\_t tsLastReport = 0;

void onBeatDetected();

int vout,val,temp;

float mv,BP;

#define CS\_PIN D8

#define CLOCK\_PIN D5

#define MOSI\_PIN D7

#define MISO\_PIN D6

MCP3008 adc(CLOCK\_PIN, MOSI\_PIN, MISO\_PIN, CS\_PIN);

const char\* ssid = "Error network";

const char\* password = "17111016";

//const char\* user\_base64 = "ZGVtb21haWxpb3RAZ21haWwuY29t";

//const char\* user\_password\_base64 = "dHJ5YWdhaW5vbmNl";

WiFiServer server(80);

void setup()

{

Serial.begin(115200);

if (!pox.begin())

for(;;);

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

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

{

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

server.begin();

Serial.println("Server started");

// Print the IP address

Serial.println(WiFi.localIP());

}

void loop()

{

pox.update();

BPM = pox.getHeartRate();

SpO2 = pox.getSpO2();

if (millis() - tsLastReport > REPORTING\_PERIOD\_MS)

{

Serial.print("BPM:");

Serial.print(BPM);

Serial.print(" SpO2 Level:");

Serial.println(SpO2);

tsLastReport = millis();

temp = adc.readADC(0)/3.1;

Serial.print("Temperature:");

Serial.println(temp);

int Val = adc.readADC(5);

Serial.print("value:");

Serial.println(val);

int BP = (val,730,740,60,160);

Serial.print("BP:");

Serial.println(BP);

}

// Check if a client has connected

WiFiClient client = server.available();

if (!client)

{

return;

}

// Wait until the client sends some data

Serial.println("new client");

while (!client.available()) {

// delay(1);

}

// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();

// Prepare the response

client.println("HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n <!DOCTYPE html> <html> <head> <title> Interfacing with NodeMCU</title> <style>");

client.println("a:link {background-color: YELLOW;text-decoration: none;}");

client.println("table, th, td {border: 1px solid black;} </style> </head> <body> <h1 style=");

client.println("font-size:300%;");

client.println(" ALIGN=CENTER> HEALTH MONITORING </h1>");

client.println("<p ALIGN=CENTER style=""font-size:150%;""");

client.println("> <b>Location Details</b></p> <table ALIGN=CENTER style=");

client.println("width:50%");

client.println("</td></tr> <tr> <th>Temperature</th> <td ALIGN=CENTER >");

client.println(temp);

client.println("</td> </tr> <tr> <th>Blood pressure</th> <td ALIGN=CENTER >");

client.println(BP);

client.println("</td> </tr> <tr> <th>Beat</th> <td ALIGN=CENTER >");

client.println(BPM);

client.println("</td> </tr> <tr> <th>spo2</th> <td ALIGN=CENTER >");

client.println(SpO2);

client.println("</td> </tr> </table> ");

client.println("<br><br>");

Serial.println("Client disconnected");

Serial.println("");

client.println("</body> </html> \n");

client.print(client);

// delay(100);

}