**FORMAN CHRISTIAN COLLEGE**

**(A CHARTERED UNIVERSITY)**



**Embedded Systems (CSCS 306)**

**FALL-2019**

**LAB-08**

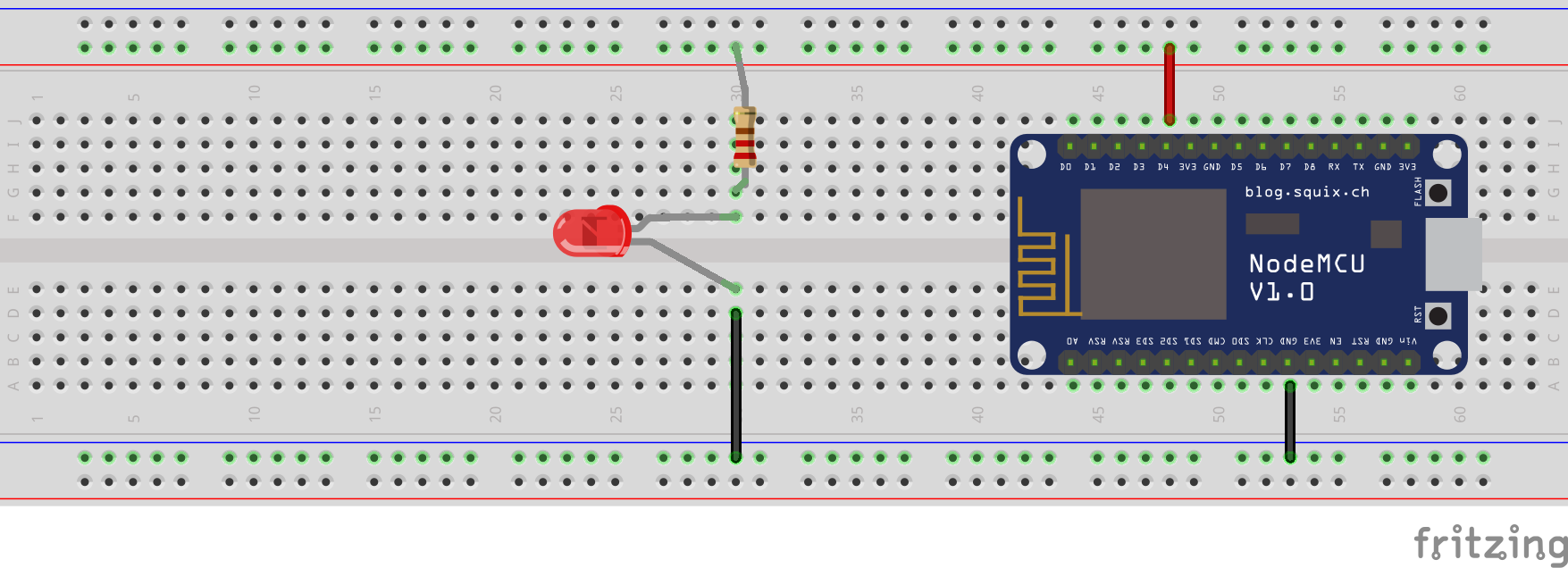
**Node MCU**

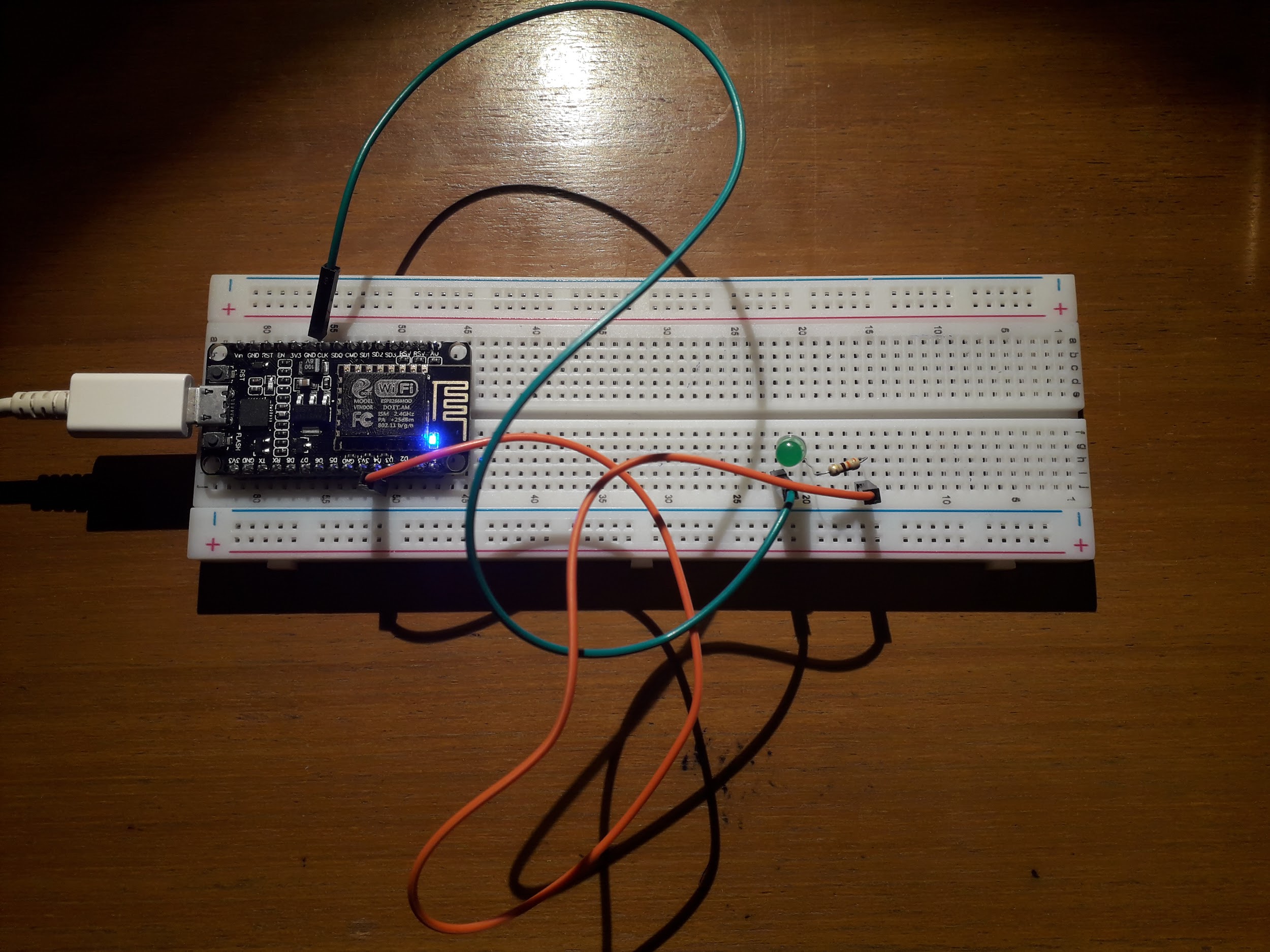
**Group Members:**

* Mohammad Usman 20-10558
* Haris Naseer 20-10612
* Abeer Butt 20-10596

**Introduction:**

The aim of the lab was to test our skills in dealing with a Node MCU. We had to display some information related to the connectivity. Make the node mcu work as a server, accept a client’s request and generate a response. In the final task, we modified the previous code so that the page displays a drop down list and a button to handle the request of the user of blinking an LED n times. This lab consists of three LabTasks.

**Circuit Diagram:Running Project Image:**

****

**LabTask 1:** In this LabTask, we had to use the ESP8266WiFi library. We connected the Node MCU to a WLAN, display the connectivity information (IP Address, MAC Address, Gateway IP) on the serial monitor.

**Working Code:**

#include <ESP8266WiFi.h>

const char \*ssid = "41bdbe";

const char \*password = "248200754";

WiFiServer server(80);

void setup()

{

Serial.begin(9600);

delay(10);

// Connect to WiFi network

Serial.println();

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");

// Start the server

server.begin();

Serial.println("Server is listening...");

// Print the IP address

Serial.print("IP Address: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/");

Serial.print("MAC Address: ");

Serial.println(WiFi.macAddress());

Serial.print("Gateway IP: ");

Serial.println(WiFi.gatewayIP());

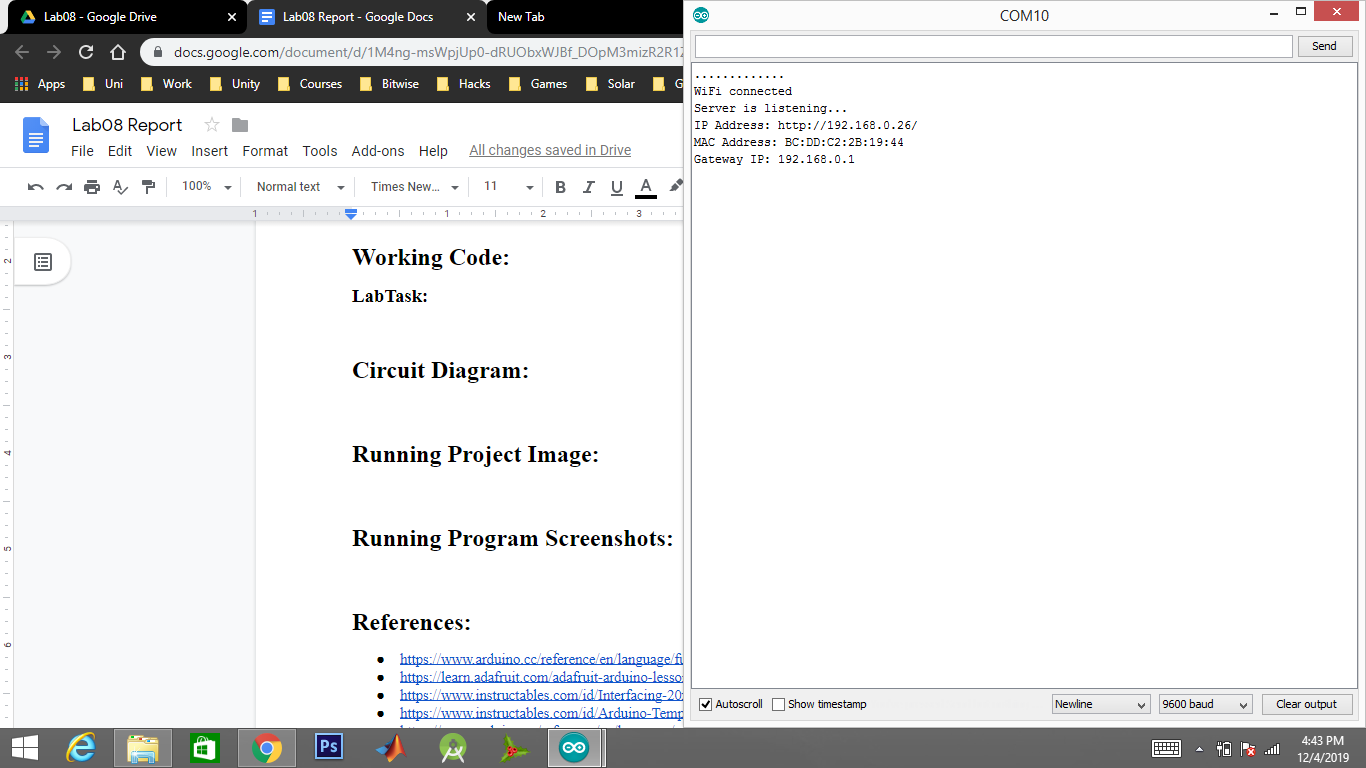
}

void loop()

{

}

**Running Program Screenshots:**

****

**LabTask 2:** In this LabTask, we make our Node MCU work as a server, made it handle a request from a browser of the client and generate an appropriate response.

**Working Code:**

#include <ESP8266WiFi.h>

const char \*ssid = "41bdbe";

const char \*password = "248200754";

int ledPin = 2; // of NodeMCU (D4 = 2)

WiFiServer server(80);

void setup()

{

Serial.begin(9600);

delay(10);

pinMode(ledPin, OUTPUT);

digitalWrite(ledPin, LOW);

// Connect to WiFi network

Serial.println();

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");

// Start the server

server.begin();

Serial.println("Server is listening...");

// Print the IP address

Serial.print("IP Address: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/");

Serial.print("MAC Address: ");

Serial.println(WiFi.macAddress());

Serial.print("Gateway IP: ");

Serial.println(WiFi.gatewayIP());

}

void loop()

{

// 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(100);

Serial.print(".");

}

Serial.println();

// Read the first line of the request

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

Serial.println(request);

client.flush();

// Match the request

int value = LOW;

if (request.indexOf("/LED=ON") != -1)

{

digitalWrite(ledPin, HIGH);

value = HIGH;

}

if (request.indexOf("/LED=OFF") != -1)

{

digitalWrite(ledPin, LOW);

value = LOW;

}

// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.print("Led is now: ");

if (value == HIGH)

{

client.print("On");

}

else

{

client.print("Off");

}

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

client.println("<a href=\"/LED=ON\">Click to ON</a>");

client.println("<a href=\"/LED=OFF\">Click to OFF</a>");

client.println("</html>");

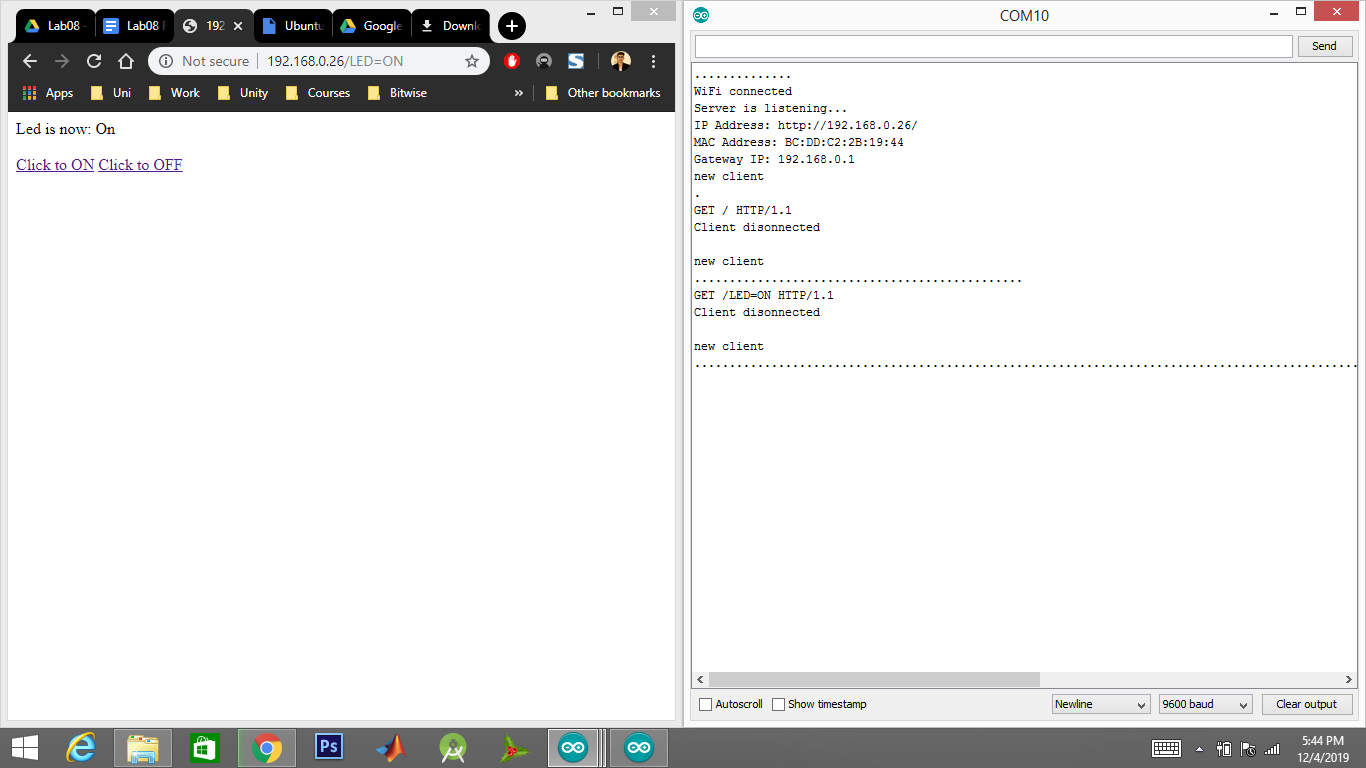
delay(1);

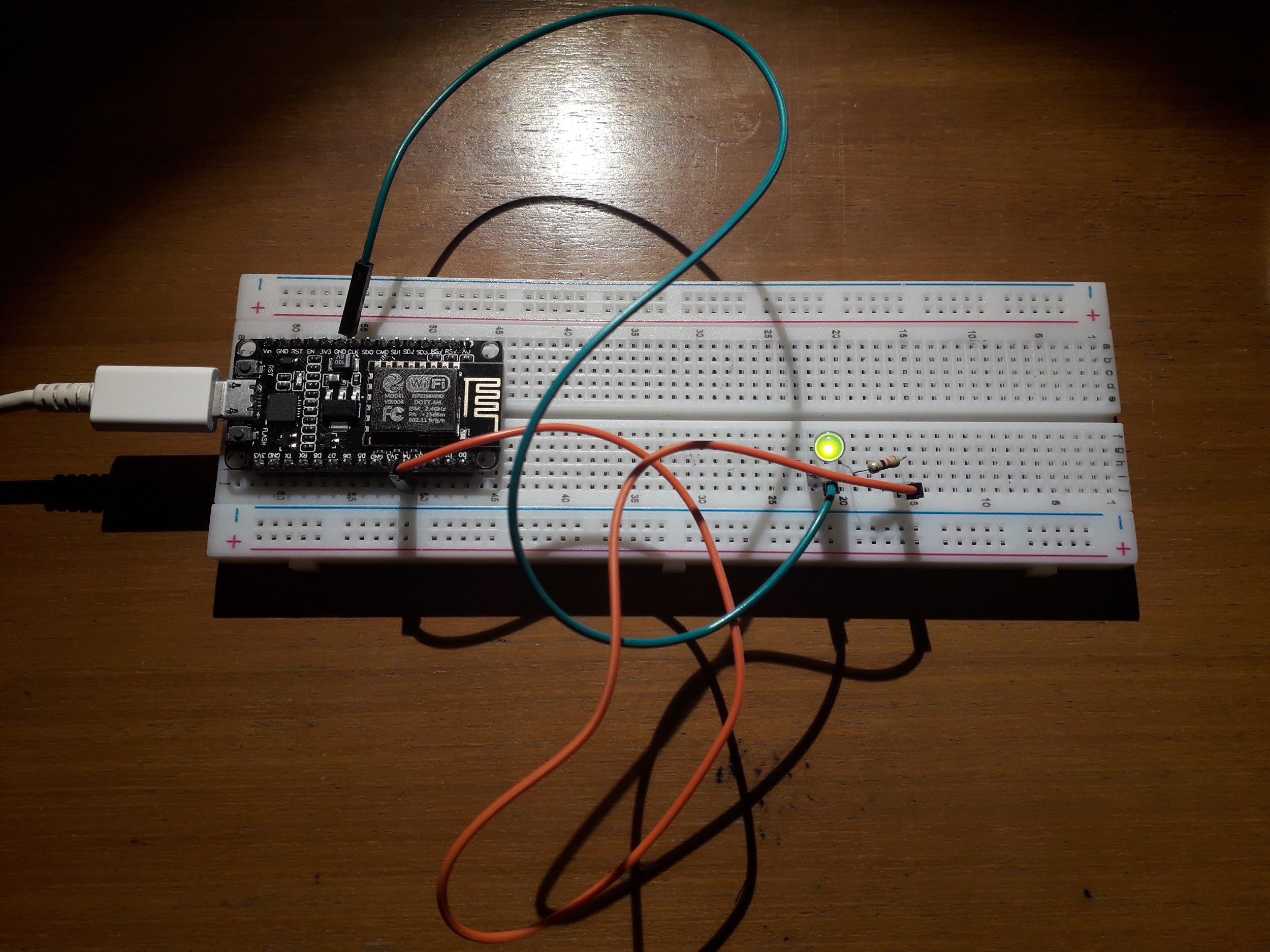
Serial.println("Client disconnected");

Serial.println("");

}

**Running Program Screenshots:**

****

****

**LabTask 3:** In this LabTask, we connected an LED with our Node MCU and made it work as a server. Displayed a drop down list and a submit button for the client to select a number from 1 to 9 which corresponds to the number of times the LED is to be blinked.

**Working Code:**

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>

String s = "<!DOCTYPE html><html><body><form action=\"/ledblink\"><select name=\"drop\"><option>Choose a Number</option><option value='1'>1</option><option value='2'>2</option><option value='3'>3</option><option value='4'>4</option><option value='5'>5</option><option value='6'>6</option><option value='7'>7</option><option value='8'>8</option><option value='9'>9</option></select><input type=\"submit\" value=\"Submit\"></form></body></html>";

const char \*ssid = "41bdbe";

const char \*password = "248200754";

ESP8266WebServer server(80); //Server on port 80

int ledPin = 2;

void handleRoot()

{

server.send(200, "text/html", s); // send web page

}

void blinkFunc(int n)

{

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

{

digitalWrite(ledPin, HIGH);

delay(1000);

digitalWrite(ledPin, LOW);

delay(1000);

}

}

void handleForm()

{

int n = server.arg("drop").toInt(); // getting input

Serial.print("LED blinking ");

Serial.print(n);

Serial.println(" times");

blinkFunc(n);

String s = "<a href='/'> Go Back </a>";

server.send(200, "text/html", s); //Send web page

}

void setup(void)

{

Serial.begin(9600);

pinMode(ledPin, OUTPUT);

digitalWrite(ledPin, LOW);

WiFi.begin(ssid, password); // connect to your WiFi router

Serial.println("");

// wait for connection

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

{

delay(500);

Serial.print(".");

}

// if connection successful show IP address in serial monitor

Serial.println("");

Serial.print("Connected to ");

Serial.println("WiFi");

Serial.print("IP address: ");

Serial.println(WiFi.localIP()); //IP address assigned to your ESP

server.on("/", handleRoot); // which routine to handle at root location

server.on("/ledblink", handleForm); // form action is handled here

server.begin(); // start server

Serial.println("HTTP server started");

}

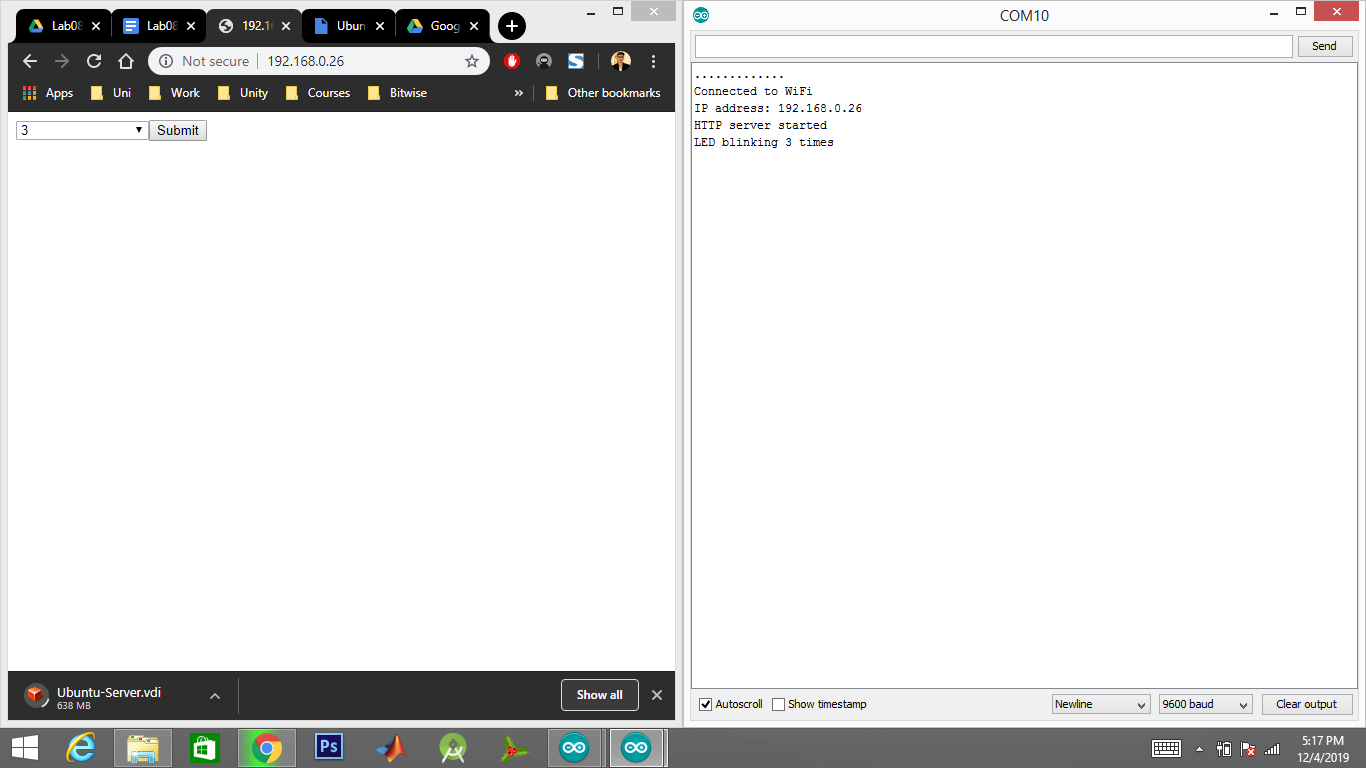
void loop(void)

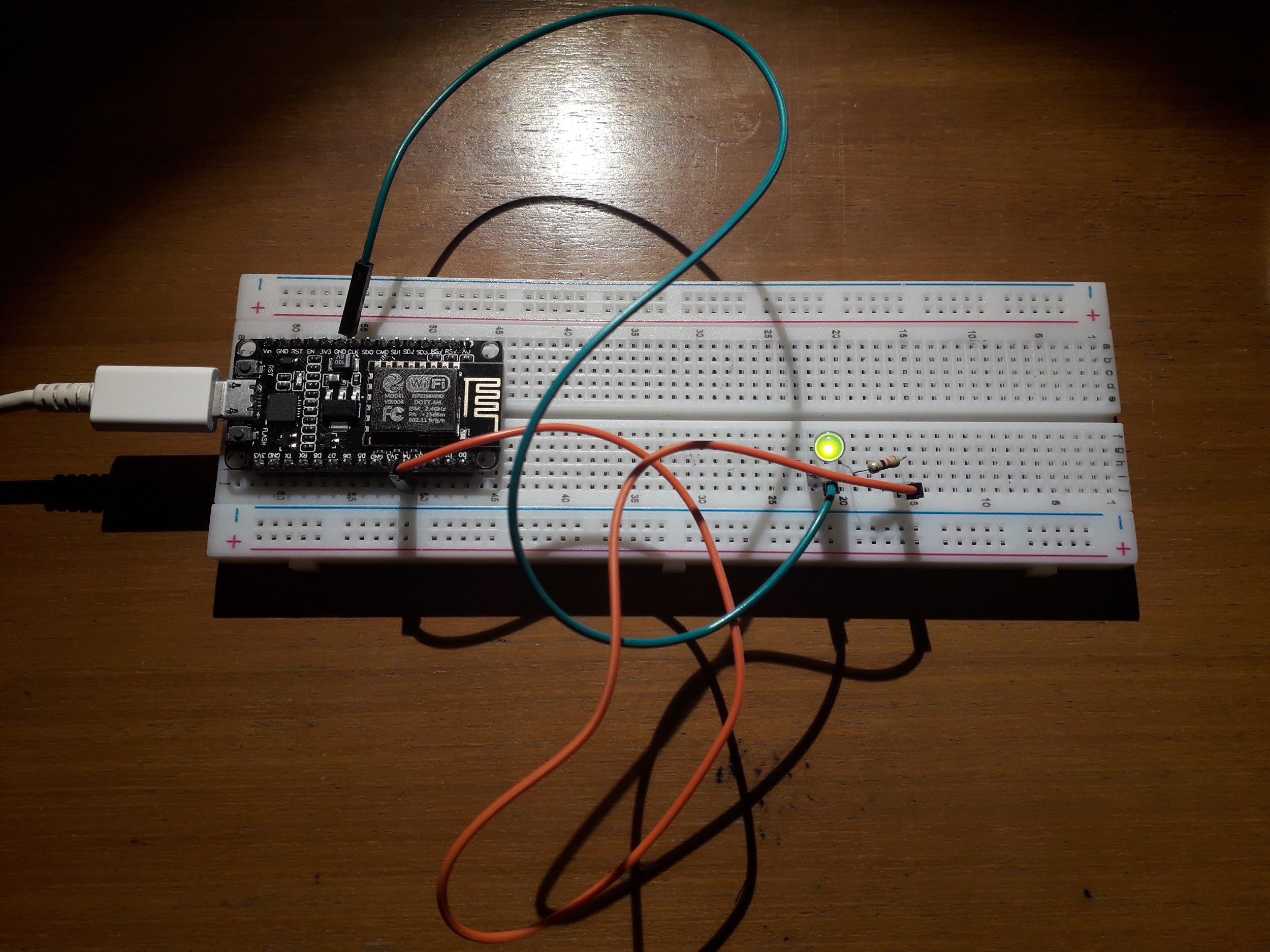
{

server.handleClient(); // handle client requests

}

**Running Program Screenshots:**

****

****

**References:**

* <https://www.arduino.cc/reference/en/language/functions/communication/serial/println/>
* <https://techtutorialsx.com/2017/04/09/esp8266-get-mac-address/>
* <https://stackoverflow.com/questions/39252057/get-gateway-address-in-arduino-using-esp8266>
* <https://www.w3schools.com/html/html_forms.asp>
* <https://www.w3schools.com/tags/tag_select.asp>
* <https://circuits4you.com/2019/03/20/esp8266-receive-post-get-request-data-from-website/>
* <https://learn.adafruit.com/adafruit-arduino-lesson-2-leds/blinking-the-led>
* <https://www.arduino.cc/en/Tutorial.StringToIntExample>
* <https://fritzing.org/>