



**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN - DCCO-SS
CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA
INFORMACIÓN**

PERIODO : octubre 2022 – marzo 2023

ASIGNATURA : Tecnologías Emergentes

TEMA : Proyecto Parcial III

NOMBRES : Ordóñez Xiomara

Pisco Lesly

NIVEL-PARALELO : Octavo

DOCENTE : Germán Rodriguez

FECHA DE ENTREGA: 6 de marzo de 2023

SANTO DOMINGO - ECUADOR

2023

DESARROLLO

Conexión a wifi

```
ConexionMQTT.ino  ConexionMQTT.ino
1 #include <ESP8266WiFi.h>
2 #include <PubSubClient.h>
3 #include <DHT.h>
4 #define DHT_PIN 0
5 #define DHT_TIPO DHT22
6
7 DHT dht(DHT_PIN, DHT_TIPO);
8 WiFiClient espClient;
9 PubSubClient client(espClient);
10
11 char msg[16];
12
13 void setup() {
14     dht.begin();
15     setup_wifi();
16     client.setServer("192.168.0.228", 1883);
17     pinMode(0, OUTPUT);
18     client.setCallback(callback);
19 }
```

Output Serial Monitor X

Message (Enter to send message to 'Generic ESP8266 Module' on 'COM3')

```
..192.168.137.226
.....Conexion exitosa
```

The screenshot shows the Arduino IDE interface. The top menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for saving, loading, and running code. The title bar displays "Generic ESP8266 Module". The main workspace shows the code for "Servo.ino". The code includes headers for WiFi and PubSubClient libraries, and defines a WiFi client and MQTT client. It also includes variables for WiFi credentials and MQTT server. The code ends with a loop that checks for messages and moves a servo based on the received values. Below the code editor are tabs for Output and Serial Monitor. The Serial Monitor window shows the program's output, starting with connection logs to a WiFi network named "xiomylesly". It then shows the WiFi connection status, IP address (192.168.137.226), and successful MQTT connection. Subsequent lines show messages being received from the MQTT topic "[servo]" with values 50, 89, and 0, -1.

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
//https://github.com/milesburton/Arduino-Temperature-Control-Library
#include <OneWire.h>
#include <Servo.h>
Servo myservo;

// Actualice con valores adecuados para su red.
const char* ssid = "xiomylesly";
const char* password = "1111111111";
const char* mqtt_server = "192.168.0.228"; //broker MQTT

WiFiClient espClient;
PubSubClient client(espClient);
long lastMsg = 0;
char msg[50];
int value = 0;

void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
  if (client.available()) {
    String msg = client.readStringUntil('\n');
    if (msg.substring(0, 7) == "servo ") {
      int value = msg.substring(7).toInt();
      myservo.write(value);
    }
  }
}
```

Message (Enter to send message to 'Generic ESP8266 Module' on 'COM3')

```
r$0$00|00010|0001?0b|?0?0?0?0?|0"??0c?0?on?1No?0?B0p00${1{lp0N0000?010?0?0?
Connecting to xiomylesly
.....
WiFi connected
IP address:
192.168.137.226
Attempting MQTT connection...connected
Message arrived [servo] 50 89
Message arrived [servo] 50 89
Message arrived [servo] 50 89
Message arrived [servo] 0 -1
Message arrived [servo] 0 -1
```

Flujo y Dashboard para el servo motor

The image displays two screenshots of a Node-RED setup on an Ubuntu system.

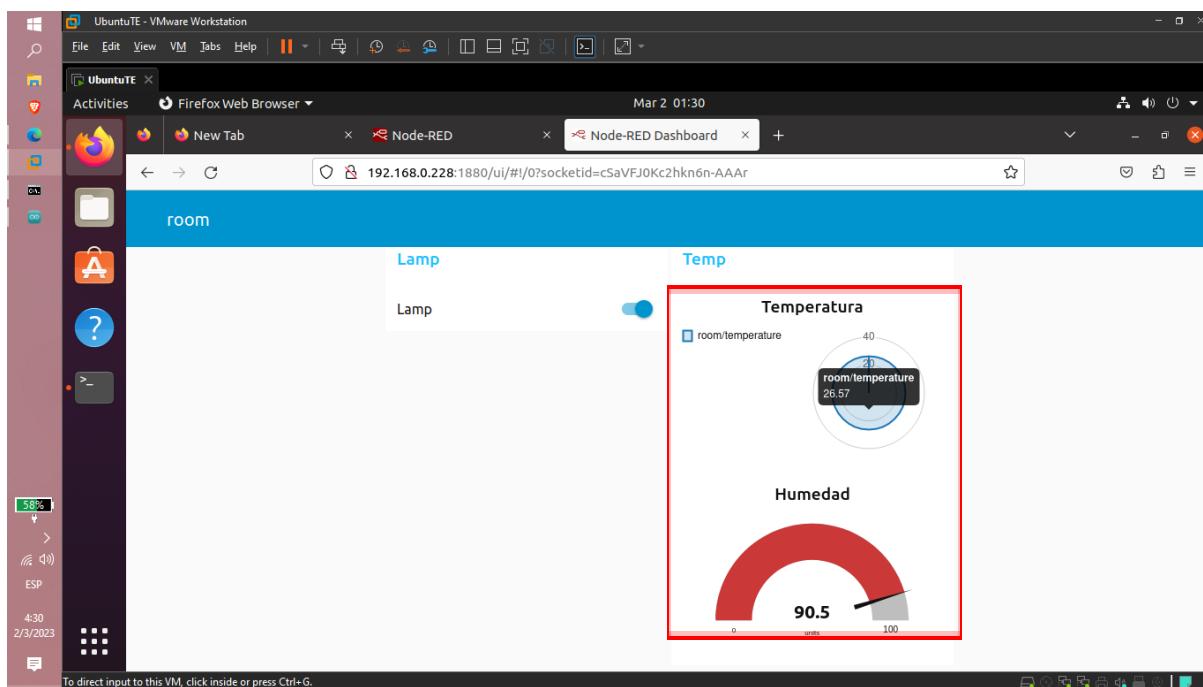
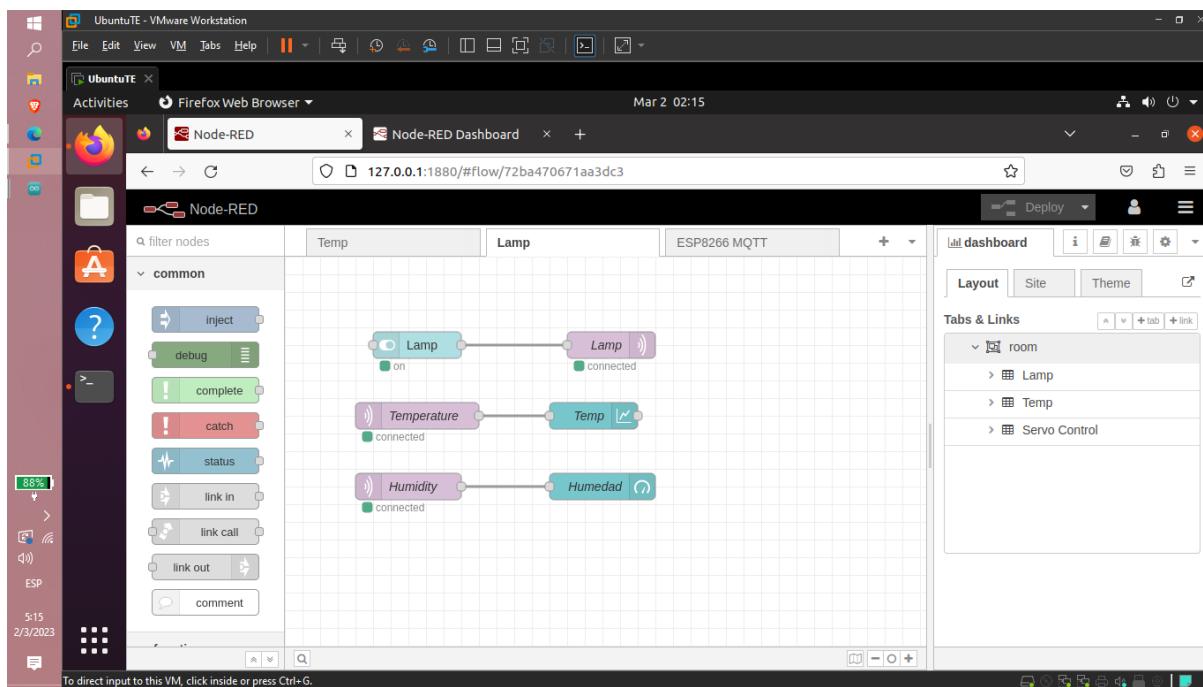
Node-RED Workspace:

- Flow 1:** A flow titled "ESP8266 MQTT Control Servo". It starts with three "button" nodes labeled "Min 0%", "50%", and "Max 100%". These buttons connect to a "node" node, which then connects to a "servo" node. The "servo" node has a status indicator showing "connected".
- Nodes Catalog:** The sidebar shows categories like "storage" (with "write file", "read file", "watch" nodes) and "dashboard" (with "button", "dropdown", "switch", "slider", "numeric" nodes).
- Info Panel:** Shows the flow ID "4e792ace.0c0c04" and a note: "ctrl + click in the workspace to open the quick-add dialog".

Browser Dashboard:

- ESP8266 SERVO_MQTT_**: A web interface titled "ESP8266" featuring a "Servo Control" section with three buttons: "MIN 0%", "50%", and "MAX 100%". Below it is a slider labeled "Servo".
- Servo position**: A circular gauge with a yellow arc indicating a value of "50" at the center.

Flujo y Dashboard para la temperatura y el led



Verificación de lectura del sensor de temperatura

sketch_mar2b | Arduino IDE 2.0.4

```

File Edit Sketch Tools Help
Generic ESP8266 Module ...
sketch_mar2b.ino
158 float hic = dht.computeHeatIndex(t, h, false);
159 static char temperatureTemp[7];
160 dtostrf(hic, 6, 2, temperatureTemp);
161
162 // Uncomment to compute temperature values in Fahrenheit
163 // float hic = dht.computeHeatIndex(f, h);
164 // static char temperatureTemp[7];
165 // dtostrf(hic, 6, 2, temperatureTemp);
166
167 static char humidityTemp[7];
168 dtostrf(h, 6, 2, humidityTemp);
169
170 // Publishes Temperature and Humidity values
171 client.publish("room/temperature", temperatureTemp);
172 client.publish("room/humidity", humidityTemp);
173
174 Serial.print("Humidity: ");
175 Serial.print(h);
176 Serial.print(" %\t Temperature: ");
177 Serial.print(t);
178 Serial.print(" *C ");
179 Serial.print(f);
180 Serial.print(" *F\t Heat index: ");
181 Serial.print(hic);

```

Output Serial Monitor x

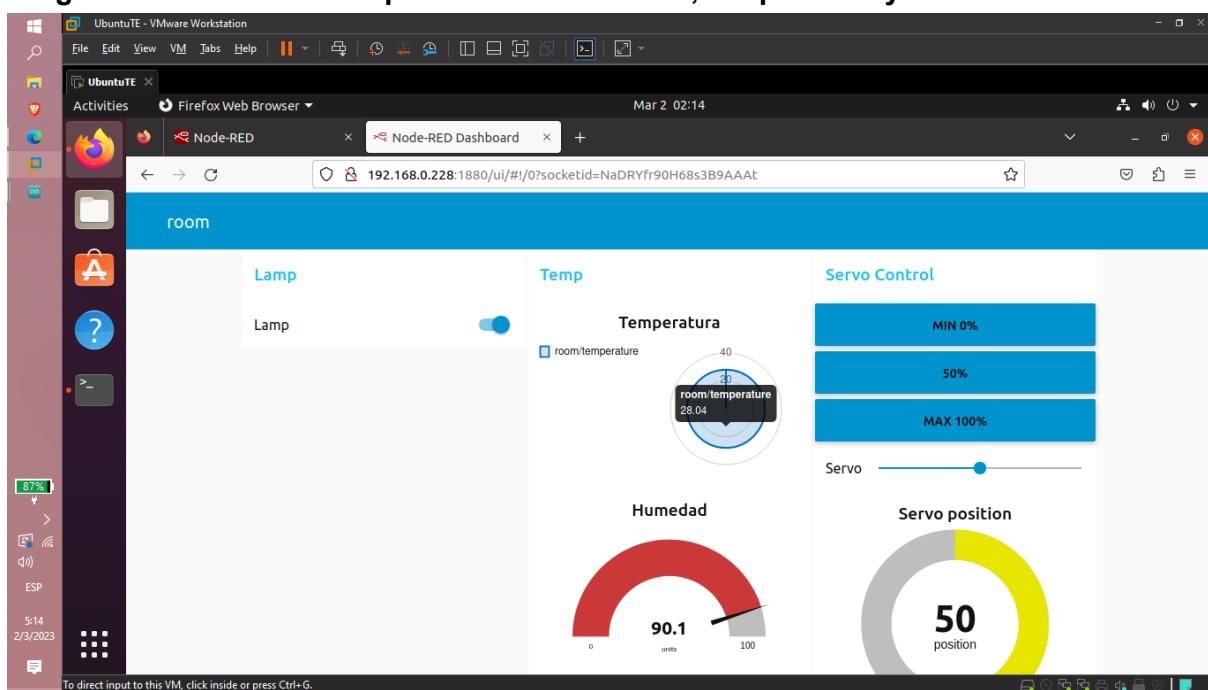
Message (Enter to send message to 'Generic ESP8266 Module' on 'COM3')

Humidity: 90.10 % Temperature: 25.50 *C 77.90 *F Heat index: 26.81 *C
Humidity: 90.30 % Temperature: 25.40 *C 77.72 *F Heat index: 26.57 *C
Humidity: 90.40 % Temperature: 25.40 *C 77.72 *F Heat index: 26.57 *C
Humidity: 90.50 % Temperature: 25.30 *C 77.54 *F Heat index: 26.33 *C
Humidity: 90.60 % Temperature: 25.30 *C 77.54 *F Heat index: 26.33 *C

New Line 115200 baud

In 169, Col 1 Generic ESP8266 Module on COM3 2

Integración de los tres componentes servo motor, temperatura y led



Arduino IDE Screenshot:

The Arduino IDE window shows a sketch named "Temp-Luz.ino". The code implements a callback function for the Generic ESP8266 Module. It prints messages received on the serial port and controls a servo and digital output based on the topic and payload.

```

Temp-Luz.ino
File Edit Sketch Tools Help
Generic ESP8266 Module
Temp-Luz.ino
void callback(String topic, byte* payload, unsigned int length) {
    Serial.print("Message arrived on topic: ");
    Serial.print(topic);
    Serial.print(". Message: ");
    String messageTemp;
    String string;
    for (int i = 0; i < length; i++) {
        Serial.print((char)payload[i]);
        messageTemp += (char)payload[i];
        string += ((char)payload[i]);
    }
    Serial.print(string);
    Serial.println();
    if (topic == "room/lamp") {
        Serial.print("Changing Room lamp to ");
        if (messageTemp == "on") {
            digitalWrite(lamp, HIGH);
            Serial.print("On");
        } else if (messageTemp == "off") {
            digitalWrite(lamp, LOW);
            Serial.print("Off");
        }
    }
}

```

Serial Monitor Output:

```

Message (Enter to send message to 'Generic ESP8266 Module' on 'COM3')
Message arrived on topic: servo. Message: 5050
Humidity: 89.60 % Temperature: 26.00 *C 78.80 *F Heat index: 28.02 *C
Message arrived on topic: room/lamp. Message: falsefalse
Changing Room lamp to
-1

```

Node-RED Dashboard Screenshot:

The Node-RED dashboard displays real-time sensor data and servo control. The "Temp" section shows a circular gauge for room temperature at 20. The "Humedad" section shows a red circular gauge for humidity at 87.8. The "Servo Control" section includes a slider for servo position and a circular gauge showing the current position at 100.

Estado de Node - Red

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@ubuntu: /home/xiomy". The log output from Node-RED shows multiple connections to an MQTT broker at 192.168.0.148:1883 and 192.168.0.228:1883, along with flow management messages like "Starting flows" and "Stopping flows".

```
2 Mar 01:58:12 - [info] [mqtt-broker:Luz Apagada] Connected to broker: mqtt://192.168.80.148:1883
2 Mar 01:58:12 - [info] [mqtt-broker:78975b76.6ffa84] Connected to broker: mqtt://192.168.0.228:1883
2 Mar 02:05:22 - [info] Stopping flows
2 Mar 02:05:22 - [info] Updated flows
2 Mar 02:05:22 - [info] Starting flows
2 Mar 02:05:22 - [info] Started flows
2 Mar 02:05:22 - [info] [mqtt-broker:Luz Apagada] Connected to broker: mqtt://192.168.80.148:1883
2 Mar 02:05:22 - [info] [mqtt-broker:78975b76.6ffa84] Connected to broker: mqtt://192.168.0.228:1883
2 Mar 02:10:53 - [info] Stopping flows
2 Mar 02:10:53 - [info] Updated flows
2 Mar 02:10:53 - [info] Starting flows
2 Mar 02:10:53 - [info] Started flows
2 Mar 02:10:53 - [info] [mqtt-broker:Luz Apagada] Connected to broker: mqtt://192.168.80.148:1883
2 Mar 02:10:53 - [info] [mqtt-broker:78975b76.6ffa84] Connected to broker: mqtt://192.168.0.228:1883
2 Mar 02:14:04 - [info] Stopping flows
2 Mar 02:14:04 - [info] Updated flows
2 Mar 02:14:04 - [info] Starting flows
2 Mar 02:14:04 - [info] Started flows
2 Mar 02:14:04 - [info] [mqtt-broker:Luz Apagada] Connected to broker: mqtt://192.168.80.148:1883
2 Mar 02:14:04 - [info] [mqtt-broker:78975b76.6ffa84] Connected to broker: mqtt://192.168.0.228:1883
```

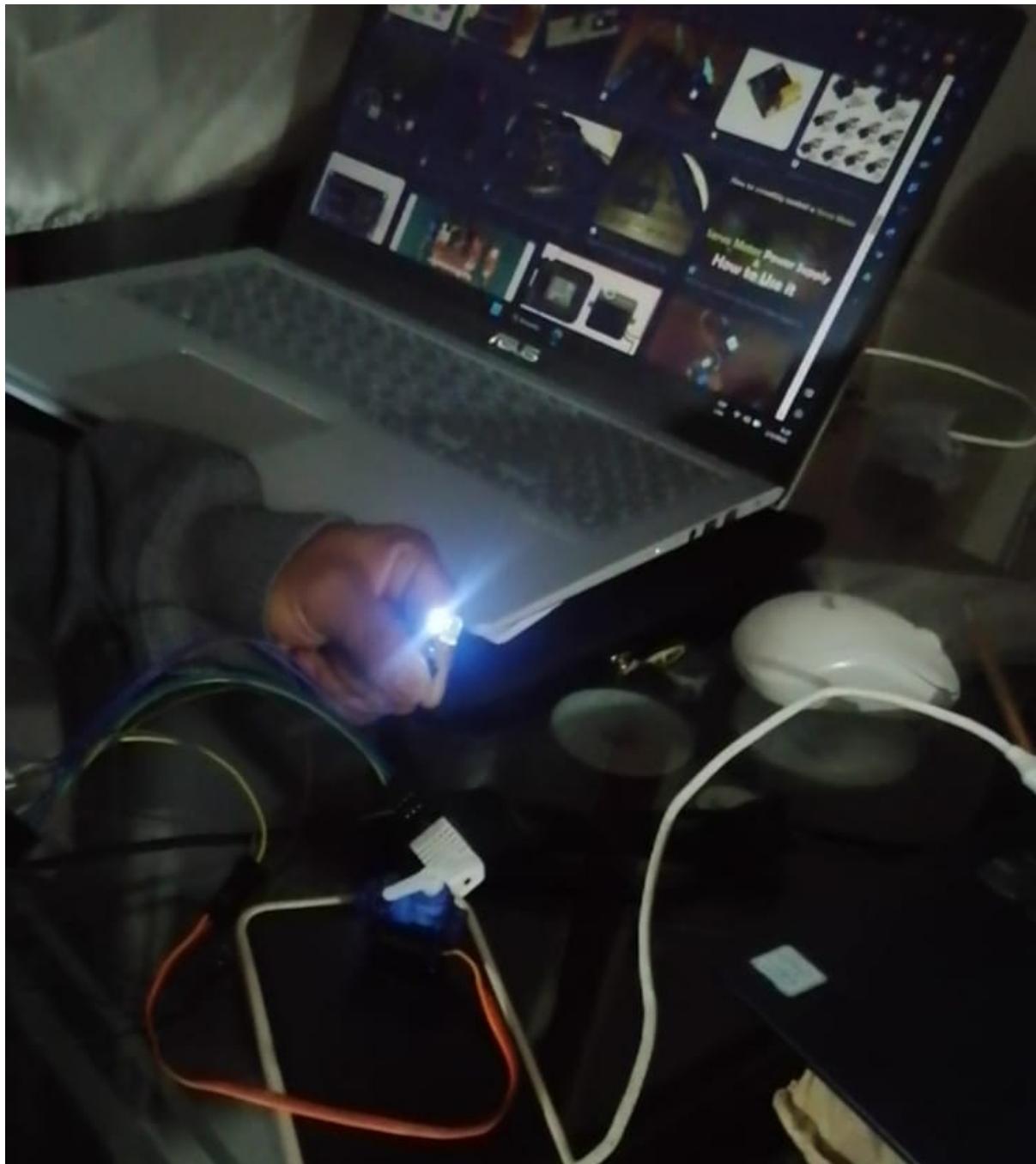
Estado de Mosquitto

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "root@ubuntu: /home/xiomy". The command run is "systemctl status mosquitto". The output shows the Mosquitto service is active and running, with a Main PID of 1004. It also lists log messages from the systemd journal.

```
root@ubuntu:/home/xiomy# systemctl status mosquitto
● mosquitto.service - Mosquitto MQTT v3.1/v3.1.1 Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-03-01 21:52:09 PST; 4h 24min ago
     Docs: man:mosquitto.conf(5)
           man:mosquitto(8)
     Main PID: 1004 (mosquitto)
        Tasks: 3 (limit: 2294)
       Memory: 1.6M
      CGroup: /system.slice/mosquitto.service
              └─1004 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

Mar 01 21:52:08 ubuntu systemd[1]: Starting Mosquitto MQTT v3.1/v3.1.1 Broker...
Mar 01 21:52:09 ubuntu mosquitto[1004]: [ - 42.272620]-DLT- 1004-INFO ~FIFO /tmp/dlt cannot be
Mar 01 21:52:09 ubuntu systemd[1]: Started Mosquitto MQTT v3.1/v3.1.1 Broker.
lines 1-14/14 (END)
```

ANEXOS



```
nt;
ent(esclient);
0;
DHTPin = 0;
onitor X
er to send message to 'Generic ESP8266 Module' on 'COM3')
-----
y: 88.60 *
e arrived on topic: room/lamp to On
Temperature: 26.40 *C 79.52 *F Heat index: 28.99 *C
-----
Temperature: 26.40 *C 79.52 *F Heat index: 28.96 *C
Temperature: 26.40 *C 79.52 *F Heat index: 28.95 *C
-----
ing
Humidity: 88.20 *
Humidity: 87.90 *
Message arrived on topic: room/lamp to Off
Changin
-1
```

