Sistemas de Comunicação Móvel, 2º Trabalho de Laboratório:

Código fonte devidamente comentado do servidor:

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

#ifndef STASSID
#define STASSID "pedmar"
#define STAPSK  "spoon123"
#endif


// Set these to your desired credentials
const char* ssid = STASSID;
const char* password = STAPSK;


// Initialize some needed variables
int temp = 0;
int hum = 0;

// Create an instance of the server
// Specify the port to listen on as an argument
ESP8266WebServer server(80);

void setup() {
  Serial.begin(115200);

  // Prepare LED
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, 0);

  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print(F("Connecting to "));
  Serial.println(ssid);

  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(F("."));
  }
  Serial.println();
  Serial.println(F("WiFi connected"));

  // Start the server
  server.on("/", handleRoot);
```

```
  server.begin();
  Serial.print("The WebServer has started on ip: ");

  // Print the IP address
  Serial.print(WiFi.localIP());
  Serial.print("\nTo operate this server, type 'temp' to update
temperature and 'hum' to update humidity!\n");
}

void loop() {
  String str;


  server.handleClient();
  Serial.setTimeout(10000);

  // Check if there are any inputs from Serial
  if (Serial.available() > 0){

    // Read new requests
    str = Serial.readStringUntil('\n');

    // Check specification of new request
    if (str.equals("temp")){
      Serial.print("\nYou have 10 seconds to introduce a value for
the temperature.");
      str = Serial.readStringUntil('\n');
      temp = str.toInt();
    } else if (str.equals("hum")){
      Serial.print("\nYou have 10 seconds to introduce a value for
the humidity.");
      str = Serial.readStringUntil('\n');
      hum = str.toInt();
    } else{
      Serial.print("\nInvalid input, please try again!");
    }
  }
}


// Manage the display of the temp and hum information on the server
void handleRoot() {
  String msg = "<h1>Welcome to the
website!</h1><br></br><br></br><h2>Temperature: </h2>";
  msg += String(temp);
  msg += "<br></br><h2>Humidity: </h2>";
  msg += String(hum);
  server.send(200, "text/html", msg);

}
```

Código fonte devidamente comentado do Access Point:

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

#ifndef APSSID
#define APSSID "pedmar"
#define APPSK  "spoon123"
#endif

// Set these to your desired credentials
const char *ssid = APSSID;
const char *password = APPSK;

// Initialize some needed variables
int prev_count = 0;
int c = 0;
String str;


// Create an instance of the server
// Specify the port to listen on as an argument
ESP8266WebServer server(80);

void setup() {
  delay(1000);
  Serial.begin(115200);
  Serial.println();
  Serial.print("Configuring access point... Please introduce a
channel for the AP.\n");

  // Configure access point
  while (true){
    // Check if there are any inputs from Serial
    while(Serial.available() > 0){
      str = Serial.readStringUntil('\n');
      c = str.toInt();
    }
    if (c < 12 && c > 0){
      break;
    }
  }
  WiFi.softAP(ssid, password, c);

  IPAddress myIP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(myIP);
  Serial.println("HTTP server started");
  WiFi.setPhyMode(WIFI_PHY_MODE_11G);
}

void loop() {
  int count;
  String variant;

  //Get number of connected stations
  count = WiFi.softAPgetStationNum();
```

```
  // Check if number of connect stations changed
  if (count != prev_count){
    Serial.print("\n*******************************\n\nChange in
number of stations connected!\n");
    Serial.print("Number of devices: ");
    Serial.println(count);
    Serial.print(F("802.11 Variant/Channel is: "));

    // Get wifi variant
    WiFiPhyMode_t mode = WiFi.getPhyMode();
    if (mode == WIFI_PHY_MODE_11B){
      variant = "802.11 b";
    } else if (mode == WIFI_PHY_MODE_11N){
      variant = "802.11 n";
    } else if (mode == WIFI_PHY_MODE_11G){
      variant = "802.11 g";
    } else {
      variant = "Invalid variant";
    }
    Serial.println(variant + F(" / ") + WiFi.channel());
    Serial.println(F("SSID used: ") + WiFi.softAPSSID());
    prev_count = count;
  }
}
```