

```

# Codi de la primera part(display):
...
#include <Arduino.h>

void init_temp_hum_task(void);

// For a connection via I2C using the Arduino Wire include:
#include <Wire.h> // Only needed for Arduino 1.6.5 and
earlier
#include "SSD1306Wire.h" // legacy: #include "SSD1306.h"
// OR #include "SH1106Wire.h" // legacy: #include "SH1106.h"

// For a connection via I2C using brzo_i2c (must be installed) include:
// #include <brzo_i2c.h> // Only needed for Arduino 1.6.5 and
earlier
// #include "SSD1306Brzo.h"
// OR #include "SH1106Brzo.h"

// For a connection via SPI include:
// #include <SPI.h> // Only needed for Arduino 1.6.5 and
earlier
// #include "SSD1306Spi.h"
// OR #include "SH1106SPi.h"

// Optionally include custom images
#include "images.h"

#include "SparkFunHTU21D.h"

//Create an instance of the object
HTU21D myHumidity;

// Initialize the OLED display using Arduino Wire:
SSD1306Wire display(0x3c, SDA, SCL); // ADDRESS, SDA, SCL - SDA and
SCL usually populate automatically based on your board's pins_arduino.h
e.g.
https://github.com/esp8266/Arduino/blob/master/variants/nodemcu/pins\_arduino.h
// SSD1306Wire display(0x3c, D3, D5); // ADDRESS, SDA, SCL - If not,
they can be specified manually.
// SSD1306Wire display(0x3c, SDA, SCL, GEOMETRY_128_32); // ADDRESS, SDA,
SCL, OLEDDISPLAY_GEOMETRY - Extra param required for 128x32 displays.
// SH1106Wire display(0x3c, SDA, SCL); // ADDRESS, SDA, SCL

// Initialize the OLED display using brzo_i2c:
// SSD1306Brzo display(0x3c, D3, D5); // ADDRESS, SDA, SCL
// or
// SH1106Brzo display(0x3c, D3, D5); // ADDRESS, SDA, SCL

// Initialize the OLED display using SPI:

```

```

// D5 -> CLK
// D7 -> MOSI (DOUT)
// D0 -> RES
// D2 -> DC
// D8 -> CS
// SSD1306Spi display(D0, D2, D8); // RES, DC, CS
// or
// SH1106Spi display(D0, D2); // RES, DC

#define DEMO_DURATION 3000
typedef void (*Demo)(void);

int demoMode = 0;
int counter = 1;

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

    Serial.println("HTU21D Example!");

    myHumidity.begin();
}

//init_temp_hum_task();
// Initialising the UI will init the display too.
display.init();

display.flipScreenVertically();
display.setFont(ArialMT_Plain_10);

}

void drawFontFaceDemo() {
    // Font Demo1
    // create more fonts at http://oleddisplay.squix.ch/
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.setFont(ArialMT_Plain_10);
    display.drawString(0, 0, "Hello world");
    display.setFont(ArialMT_Plain_16);
    display.drawString(0, 10, "Hello world");
    display.setFont(ArialMT_Plain_24);
    display.drawString(0, 26, "Hello world");
}

void drawTextFlowDemo() {
    display.setFont(ArialMT_Plain_10);
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.drawStringMaxWidth(0, 0, 128,
        "Lorem ipsum\n dolor sit amet, consetetur
s adipscing elit, sed diam nonumy eirmod tempor invidunt ut labore." );
}

```

```

void drawTextAlignmentDemo() {
    // Text alignment demo
    display.setFont(ArialMT_Plain_10);

    // The coordinates define the left starting point of the text
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.drawString(0, 10, "Left aligned (0,10)");

    // The coordinates define the center of the text
    display.setTextAlignment(TEXT_ALIGN_CENTER);
    display.drawString(64, 22, "Center aligned (64,22)");

    // The coordinates define the right end of the text
    display.setTextAlignment(TEXT_ALIGN_RIGHT);
    display.drawString(128, 33, "Right aligned (128,33)");
}

void drawRectDemo() {
    // Draw a pixel at given position
    for (int i = 0; i < 10; i++) {
        display.setPixel(i, i);
        display.setPixel(10 - i, i);
    }
    display.drawRect(12, 12, 20, 20);

    // Fill the rectangle
    display.fillRect(14, 14, 17, 17);

    // Draw a line horizontally
    display.drawHorizontalLine(0, 40, 20);

    // Draw a line horizontally
    display.drawVerticalLine(40, 0, 20);
}

void drawCircleDemo() {
    for (int i = 1; i < 8; i++) {
        display.setColor(WHITE);
        display.drawCircle(32, 32, i * 3);
        if (i % 2 == 0) {
            display.setColor(BLACK);
        }
        display.fillCircle(96, 32, 32 - i * 3);
    }
}

void drawProgressBarDemo() {
    int progress = (counter / 5) % 100;
    // draw the progress bar
    display.drawProgressBar(0, 32, 120, 10, progress);

    // draw the percentage as String
    display.setTextAlignment(TEXT_ALIGN_CENTER);

```

```

    display.drawString(64, 15, String(progress) + "%");
}

void drawImageDemo() {
    // see http://blog.squix.org/2015/05/esp8266-nodemcu-how-to-create-xbm.html
    // on how to create xbm files
    display.drawXbm(34, 14, WiFi_Logo_width, WiFi_Logo_height,
WiFi_Logo_bits);
}

Demo demos[] = {drawFontFaceDemo, drawTextFlowDemo, drawTextAlignmentDemo,
drawRectDemo, drawCircleDemo, drawProgressBarDemo, drawImageDemo};
int demoLength = (sizeof(demos) / sizeof(Demo));
long timeSinceLastModeSwitch = 0;

void loop() {

    float humd = myHumidity.readHumidity();
    float temp = myHumidity.readTemperature();

    Serial.print("Time:");
    Serial.print(millis());
    Serial.print(" Temperature:");
    Serial.print(temp, 1);
    Serial.print("C");
    Serial.print(" Humidity:");
    Serial.print(humd, 1);
    Serial.print("%");

    Serial.println();

    // clear the display
    display.clear();
    // draw the current demo method
    //demos[demoMode]();

    display.setTextAlignment(TEXT_ALIGN_CENTER);
    display.setFont(ArialMT_Plain_10);
    display.drawString(128/2, 0, "HUMEDAD");
    display.setFont(ArialMT_Plain_16);
    display.drawString(128/2, 11, String(humd) + "%");
    display.setFont(ArialMT_Plain_10);
    display.drawString(128/2, 30, "TEMPERATURA");
    display.setFont(ArialMT_Plain_16);
    display.drawString(128/2, 41, String(temp) + "Â°C");

    display.setFont(ArialMT_Plain_10);
    display.setTextAlignment(TEXT_ALIGN_RIGHT);
    display.drawString(128, 54, String(millis()/3600000)+String(":")\
        +String((millis()/60000)%60)+String(":")\
        +String((millis()/1000)%60));
}

```

```

    // write the buffer to the display
    display.display();

    delay(100);
}

...

### Explicaci3 del codi:

Primer de tot, tenim el setup, el qual s'encarrega de inicialitzar el
display amb els prints necess3ris.
...
void setup() {
    Serial.begin(115200);
    Serial.println();
    Serial.println();

    Serial.println("HTU21D Example!");

    myHumidity.begin();
//}

    //init_temp_hum_task();
    // Initialising the UI will init the display too.
    display.init();

    display.flipScreenVertically();
    display.setFont(ArialMT_Plain_10);

}
...

```

A continuaci3 les funcions void que indiquen tot tipus de dades dins del display, encara que en el cas del nostre display no podrem utilitzar la majoria d'aquestes funcions degut a que es un model bastant limitat. Nom3s treu dades en blanc i negre.

```

...
void drawFontFaceDemo() {
    // Font Demo1
    // create more fonts at http://oleddisplay.squix.ch/
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.setFont(ArialMT_Plain_10);
    display.drawString(0, 0, "Hello world");
    display.setFont(ArialMT_Plain_16);
    display.drawString(0, 10, "Hello world");
    display.setFont(ArialMT_Plain_24);
}

```

```

    display.drawString(0, 26, "Hello world");
}

void drawTextFlowDemo() {
    display.setFont(ArialMT_Plain_10);
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.drawStringMaxWidth(0, 0, 128,
        "Lorem ipsum\n dolor sit amet, consetetur
s adipscing elit, sed diam nonumy eirmod tempor invidunt ut labore." );
}

void drawTextAlignmentDemo() {
    // Text alignment demo
    display.setFont(ArialMT_Plain_10);

    // The coordinates define the left starting point of the text
    display.setTextAlignment(TEXT_ALIGN_LEFT);
    display.drawString(0, 10, "Left aligned (0,10)");

    // The coordinates define the center of the text
    display.setTextAlignment(TEXT_ALIGN_CENTER);
    display.drawString(64, 22, "Center aligned (64,22)");

    // The coordinates define the right end of the text
    display.setTextAlignment(TEXT_ALIGN_RIGHT);
    display.drawString(128, 33, "Right aligned (128,33)");
}

void drawRectDemo() {
    // Draw a pixel at given position
    for (int i = 0; i < 10; i++) {
        display.setPixel(i, i);
        display.setPixel(10 - i, i);
    }
    display.drawRect(12, 12, 20, 20);

    // Fill the rectangle
    display.fillRect(14, 14, 17, 17);

    // Draw a line horizontally
    display.drawHorizontalLine(0, 40, 20);

    // Draw a line vertically
    display.drawVerticalLine(40, 0, 20);
}

void drawCircleDemo() {
    for (int i = 1; i < 8; i++) {
        display.setColor(WHITE);
        display.drawCircle(32, 32, i * 3);
        if (i % 2 == 0) {
            display.setColor(BLACK);
        }
        display.fillCircle(32, 32, i * 3);
    }
}

```

```

    }
}

void drawProgressBarDemo() {
    int progress = (counter / 5) % 100;
    // draw the progress bar
    display.drawProgressBar(0, 32, 120, 10, progress);

    // draw the percentage as String
    display.setTextAlignment(TEXT_ALIGN_CENTER);
    display.drawString(64, 15, String(progress) + "%");
}

void drawImageDemo() {
    // see http://blog.squix.org/2015/05/esp8266-nodemcu-how-to-create-xbm.html
    // on how to create xbm files
    display.drawXbm(34, 14, WiFi_Logo_width, WiFi_Logo_height,
WiFi_Logo_bits);
}

Demo demos[] = {drawFontFaceDemo, drawTextFlowDemo, drawTextAlignmentDemo,
drawRectDemo, drawCircleDemo, drawProgressBarDemo, drawImageDemo};
int demoLength = (sizeof(demos) / sizeof(Demo));
long timeSinceLastModeSwitch = 0;
...

Per Ãltim tenim el loop, el qual farÃ possible mostrar per pantalla del
diplay totes les dades necessÃries.

...

void loop() {

    float humd = myHumidity.readHumidity();
    float temp = myHumidity.readTemperature();

    Serial.print("Time:");
    Serial.print(millis());
    Serial.print(" Temperature:");
    Serial.print(temp, 1);
    Serial.print("C");
    Serial.print(" Humidity:");
    Serial.print(humd, 1);
    Serial.print("%");

    Serial.println();

    // clear the display
    display.clear();
    // draw the current demo method
    //demos[demoMode]();

    display.setTextAlignment(TEXT_ALIGN_CENTER);

```

```

display.setFont(ArialMT_Plain_10);
display.drawString(128/2, 0, "HUMEDAD");
display.setFont(ArialMT_Plain_16);
display.drawString(128/2, 11, String(humd)+ "%");
display.setFont(ArialMT_Plain_10);
display.drawString(128/2, 30, "TEMPERATURA");
display.setFont(ArialMT_Plain_16);
display.drawString(128/2, 41, String(temp)+ "°C");

display.setFont(ArialMT_Plain_10);
display.setTextAlignment(TEXT_ALIGN_RIGHT);
display.drawString(128, 54, String(millis()/3600000)+String(":")\
    +String((millis()/60000)%60)+String(":")\
    +String((millis()/1000)%(60)));

// write the buffer to the display
display.display();

delay(100);
}

...

# Codi de la segona part (WEB):

# Pràctica 5 Web:
### CÀlculo:

...

/*****
  Rui Santos
  Complete project details at https://randomnerdtutorials.com
*****/

// Import required libraries
#include "WiFi.h"
#include "ESPAsyncWebServer.h"
#include <Adafruit_Sensor.h>
#include "SparkFunHTU21D.h"
#include <Wire.h>
#include "SSD1306Wire.h"

// Replace with your network credentials
const char* ssid = "Xiaomi_11T_Pro";
const char* password = "f5cbd8a82232";

// Digital pin connected to the DHT sensor

// Uncomment the type of sensor in use:

```



```

// #define DHTTYPE      DHT11      // DHT 11      // DHT 22 (AM2302)
// #define DHTTYPE      DHT21      // DHT 21 (AM2301)

float temp = 0;
float humd = 0;

HTU21D myHumidity;

SSD1306Wire display(0x3c, SDA, SCL);

// Create AsyncWebServer object on port 80
AsyncWebServer server(80);

String readmyHumiditytemperature() {
    // Sensor readings may also be up to 2 seconds 'old' (its a very slow
    sensor)
    // Read temperature as Celsius (the default)
    float t = temp;
    // Read temperature as Fahrenheit (isFahrenheit = true)
    // float t = dht.readTemperature(true);
    // Check if any reads failed and exit early (to try again).
    if (isnan(t)) {
        Serial.println("Failed to read from DHT sensor!");
        return "--";
    }
    else {
        Serial.println(t);
        return String(t);
    }
}

String readmyHumidityHumidity() {
    // Sensor readings may also be up to 2 seconds 'old' (its a very slow
    sensor)
    float h = humd;
    if (isnan(h)) {
        Serial.println("Failed to read from DHT sensor!");
        return "--";
    }
    else {
        Serial.println(h);
        return String(h);
    }
}

const char index_html[] PROGMEM = R"rawliteral(
<!DOCTYPE HTML><html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css"
integrity="sha384-

```

```

fnmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9Sr"
crossorigin="anonymous">
<style>
  html {
    font-family: Arial;
    display: inline-block;
    margin: 0px auto;
    text-align: center;
  }
  h2 { font-size: 3.0rem; }
  p { font-size: 3.0rem; }
  .units { font-size: 1.2rem; }
  .dht-labels{
    font-size: 1.5rem;
    vertical-align:middle;
    padding-bottom: 15px;
  }
</style>
</head>
<body>
  <h2>ESP32 DHT Server</h2>
  <p>
    <i class="fas fa-thermometer-half" style="color:#059e8a;"></i>
    <span class="dht-labels">Temperature</span>
    <span id="temperature">%TEMPERATURE%</span>
    <sup class="units">&deg;C</sup>
  </p>
  <p>
    <i class="fas fa-tint" style="color:#00add6;"></i>
    <span class="dht-labels">Humidity</span>
    <span id="humidity">%HUMIDITY%</span>
    <sup class="units">&percnt;</sup>
  </p>
</body>
<script>
setInterval(function ( ) {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("temperature").innerHTML =
this.responseText;
    }
  };
  xhttp.open("GET", "/temperature", true);
  xhttp.send();
}, 10000 ) ;

setInterval(function ( ) {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("humidity").innerHTML = this.responseText;
    }
  };
},

```

```

    xhttp.open("GET", "/humidity", true);
    xhttp.send();
}, 10000 ) ;
</script>
</html>rawliteral";

// Replaces placeholder with DHT values
String processor(const String& var){
  //Serial.println(var);
  if(var == "TEMPERATURE"){
    return readmyHumiditytemperature();
  }
  else if(var == "HUMIDITY"){
    return readmyHumidityHumidity();
  }
  return String();
}

void setup(){

  display.init();

  display.flipScreenVertically();
  display.setFont(ArialMT_Plain_10);

  // Serial port for debugging purposes
  Serial.begin(115200);

  myHumidity.begin();

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi..");
  }

  // Print ESP32 Local IP Address
  Serial.println(WiFi.localIP());

  // Route for root / web page
  server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
    request->send_P(200, "text/html", index_html, processor);
  });
  server.on("/temperature", HTTP_GET, [](AsyncWebServerRequest *request){
    request->send_P(200, "text/plain",
readmyHumiditytemperature().c_str());
  });
  server.on("/humidity", HTTP_GET, [](AsyncWebServerRequest *request){
    request->send_P(200, "text/plain", readmyHumidityHumidity().c_str());
  });

  // Start server
  server.begin();

```

```

}

void loop(){
  humd = myHumidity.readHumidity();
  temp = myHumidity.readTemperature();

  Serial.print("Time:");
  Serial.print(millis());
  Serial.print(" Temperature:");
  Serial.print(temp, 1);
  Serial.print("C");
  Serial.print(" Humidity:");
  Serial.print(humd, 1);
  Serial.print("%");

  Serial.println();

  // clear the display
  display.clear();
  // draw the current demo method
  //demos[demoMode]();

  display.setTextAlignment(TEXT_ALIGN_CENTER);
  display.setFont(ArialMT_Plain_10);
  display.drawString(128/2, 0, "HUMEDAD");
  display.setFont(ArialMT_Plain_16);
  display.drawString(128/2, 11, String(humd) + "%");
  display.setFont(ArialMT_Plain_10);
  display.drawString(128/2, 30, "TEMPERATURA");
  display.setFont(ArialMT_Plain_16);
  display.drawString(128/2, 41, String(temp) + "°C");

  display.setFont(ArialMT_Plain_10);
  display.setTextAlignment(TEXT_ALIGN_RIGHT);
  display.drawString(128, 54, String(millis()/3600000)+String(":")\
    +String((millis()/60000)%60)+String(":")\
    +String((millis()/1000)%60));

  // write the buffer to the display
  display.display();

  delay(100);
}
...

```

Explicaci3 del codi:

Aquest codi és molt similar a la part del display amb la funció de diferència de la part de la pàgina web, per tant a continuació es mostra el codi HTML de la web.

```
<!DOCTYPE HTML><html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css"
integrity="sha384-
fmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9Sr"
crossorigin="anonymous">
  <style>
    html {
      font-family: Arial;
      display: inline-block;
      margin: 0px auto;
      text-align: center;
    }
    h2 { font-size: 3.0rem; }
    p { font-size: 3.0rem; }
    .units { font-size: 1.2rem; }
    .dht-labels{
      font-size: 1.5rem;
      vertical-align:middle;
      padding-bottom: 15px;
    }
  </style>
</head>
<body>
  <h2>ESP32 DHT Server</h2>
  <p>
    <i class="fas fa-thermometer-half" style="color:#059e8a;"></i>
    <span class="dht-labels">Temperature</span>
    <span id="temperature">%TEMPERATURE%</span>
    <sup class="units">&deg;C</sup>
  </p>
  <p>
    <i class="fas fa-tint" style="color:#00add6;"></i>
    <span class="dht-labels">Humidity</span>
    <span id="humidity">%HUMIDITY%</span>
    <sup class="units">&percnt;</sup>
  </p>
</body>
<script>
setInterval(function ( ) {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("temperature").innerHTML =
this.responseText;
    }
  };
  xhttp.open("GET", "/temperature", true);
```

```

    xhttp.send();
}, 10000 ) ;

setInterval(function ( ) {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("humidity").innerHTML = this.responseText;
        }
    };
    xhttp.open("GET", "/humidity", true);
    xhttp.send();
}, 10000 ) ;
</script>
</html>rawliteral";

```

...

Per tal de mostrar la IP de la pàgina web caldrà connectar el ESP32 a una xarxa Wifi, tal com es mostra a continuació³.

...

```

// Replace with your network credentials
const char* ssid = "Xiaomi_11T_Pro";
const char* password = "f5cbd8a82232";

```

...