

Home Automation using Arduino Cloud

Name: Santiago Cruz

Date: 04/10/2024

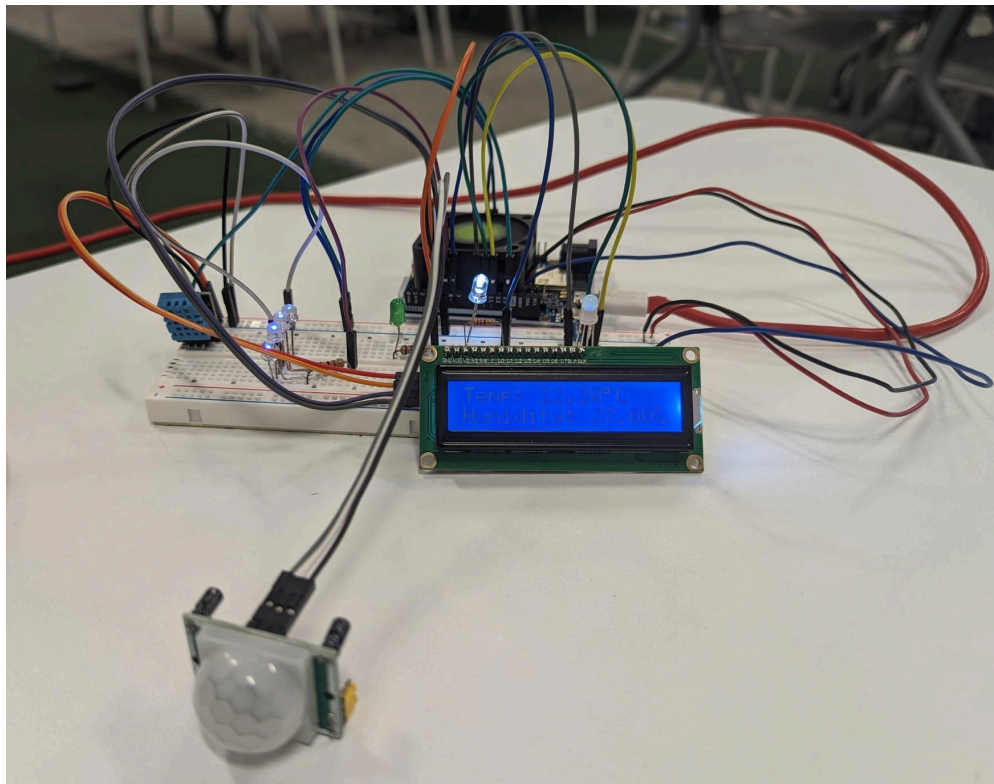
This Home Automation system combines the power of Arduino Cloud with various sensors and LEDs to offer features like motion detection, temperature and humidity monitoring, and customizable lighting control, here are some highlights:

Enhanced security: Receive alerts on your phone whenever the PIR sensor detects motion, for improved peace of mind.

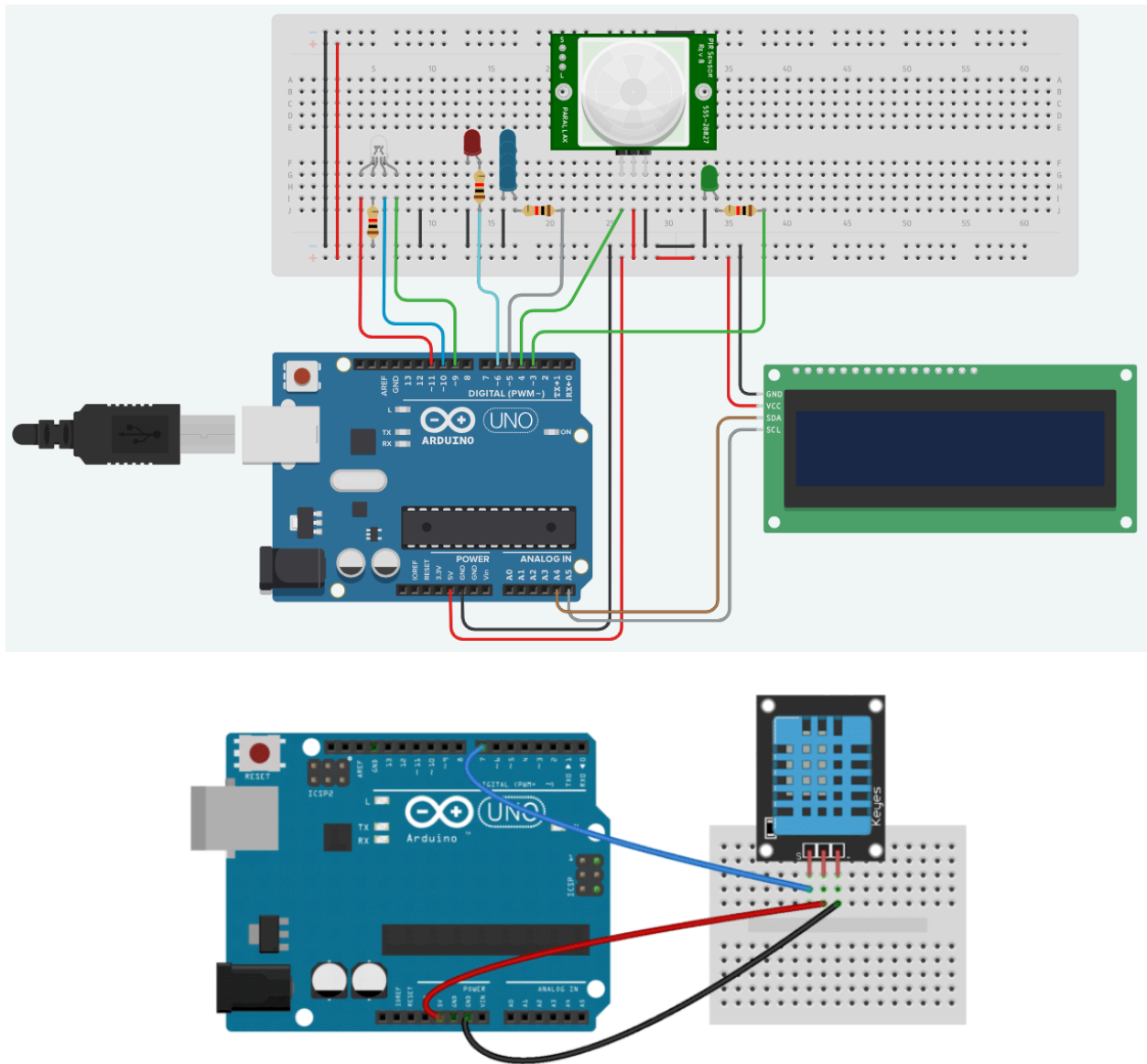
Real-time monitoring: Keep an eye on your home's temperature and humidity with the DHT11 sensor, accessible remotely through Arduino Cloud.

Customizable ambiance: Set the mood with the RGB LED and control individual blue, and red LEDs to create personalized lighting schemes.

1. Arduino Diagram



Thinkercad Link: <https://www.tinkercad.com/things/3Prs292AaEp-iot-final-project>



2. Components and supplies

- Arduino UNO R4 WIFI
- IC2 Liquid Crystal Display
- DHT11
- 3 x 1K Ω
- 220 Ω
- RGB Led
- 4 x Blue Led, One White Led, and One Red Led

- HC-SR501 PIR Motion
- 330 Ω
- Green Led

3. Connecting Arduino UNO R4 WIFI to Arduino Cloud

Once we have connected the Arduino to the PC and Arduino Cloud, we will create a **"Thing"** and then connect the Arduino to the network, finally, we can modify the sketch and add variables.

The screenshot shows the Arduino Cloud 'Setup' tab for a new 'Thing' named 'IoT Final Project'. The interface includes a 'Cloud Variables' section with an 'ADD' button and a description: 'Variables are what you can monitor or control to make your Thing function. For example a temperature or a smart lamp. Once created, you can use them in your Sketch.' Below this is a diagram of a computer monitor with various icons (Wi-Fi, cloud, lightbulb, etc.) around it. To the right, the 'Associated Device' section shows 'Arduino-R4-WiFi' with its ID, type, and status (Offline). There are 'Change' and 'Detach' buttons. The 'Network' section prompts the user to 'Enter your network credentials to connect your device.' with a 'Configure' button.

Make sure the Arduino UNO R4 WiFi is connected to our network:

Configure network

×

Enter your network credentials to allow your device to connect to the Cloud.

Wi-Fi Name *

IT Administrator

Password

.....

👁

IMPORTANT: Remember to go to the **"Sketch"** tab and upload the sketch to load the credentials on the board.

SAVE

4. Creating the Thing and Dashboard

Thing
IoT - Final Project

Setup Sketch

Cloud Variables ADD

Name ↓	Last Value	Last Update	
<input type="checkbox"/> blue_led_switch <code>bool blue_led_switch;</code>	true	10 Apr 2024 09:14:57	⋮
<input type="checkbox"/> humidity <code>CloudRelativeHumidity humidity;</code>	21	08 Apr 2024 17:52:38	⋮
<input type="checkbox"/> motion_thing <code>bool motion_thing;</code>	-		⋮
<input type="checkbox"/> red_switch <code>CloudDimmedLight red_switch;</code>	{"bri":41,"swi":"tru...	10 Apr 2024 09:15:03	⋮
<input type="checkbox"/> temp <code>CloudTemperatureSensor temp;</code>	23.2	08 Apr 2024 17:58:17	⋮

Associated Device

Arduino UNO R4
WiFi

ID: 7e71b6dc-1488-4f18-b2ef-6...
Type: Arduino UNO R4 WiFi
Status: ● Online

Change Detach

Network

Wi-Fi Name: IT-Admi...
Password:
Change

After adding the cloud variables and ensuring the status of the Arduino board is **Online**, only after that can we begin to create the dashboard for the Thing.

- Desktop Layout:

Dashboard
IoT - Final Project

ADD + 📱

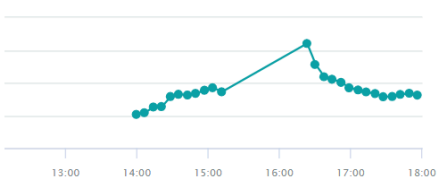
Blue Led Switch
ON

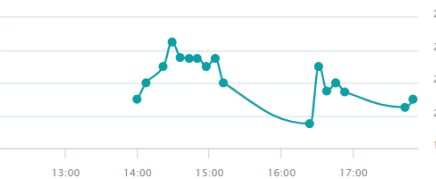
Motion Detection
TRUE

Red Dimmed light
ON
53 %

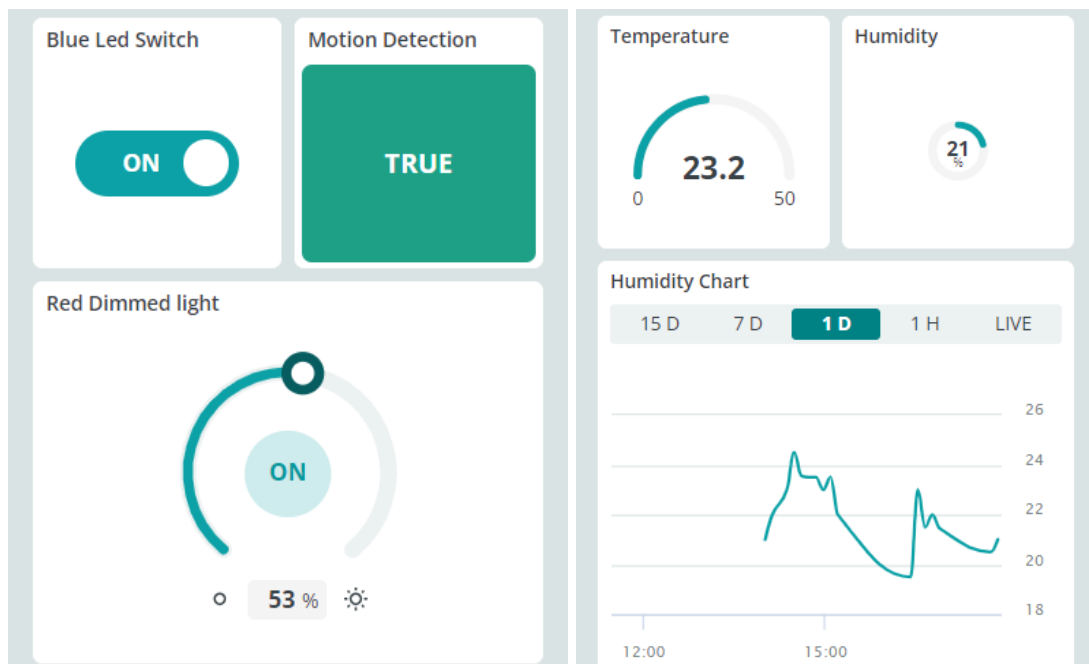
Temperature
23.2
0 50

Humidity
21 %

Temperature Chart
15 D 7 D 1 D 1 H LIVE


Humidity Chart
15 D 7 D 1 D 1 H LIVE


- Mobile Layout:



5. Arduino Code

```

/*
  Sketch generated by the Arduino IoT Cloud Thing "Untitled"

https://create.arduino.cc/cloud/things/36ab5049-8455-48cd-82ed-8ea5e1ccd8b1

  Arduino IoT Cloud Variables description

  The following variables are automatically generated and updated
  when changes are made to the Thing

  CloudDimmedLight red_switch;
  CloudTemperatureSensor temp;
  CloudRelativeHumidity humidity;
  bool blue_led_switch;
  bool motion_thing;

  Variables which are marked as READ/WRITE in the Cloud Thing will
  also have functions

```

```
    which are called when their values are changed from the Dashboard.  
    These functions are generated with the Thing and added at the end  
of this sketch.  
*/  
  
#include <DHT.h>  
#include <DHT_U.h>  
  
#include <LCD_I2C.h>  
  
#include "thingProperties.h"  
  
#define dht_sensor_pin 13  
#define dht_sensor_type DHT11  
  
DHT dht_sensor(dht_sensor_pin, dht_sensor_type);  
  
LCD_I2C lcd(0x27, 16, 2);  
  
//PIR SENSOR  
int pir_sensor_pin = 4;  
int led_pir = 3;  
int val_pir = 0;  
  
//RGB LED, PWM LED AND BLUE LEDS  
int LED_PWM_PIN = 6;  
int led_pin1b = 5;  
int rgbR = 9;  
int rgbG = 10;  
int rgbB = 11;  
  
void setup() {  
    // Initialize serial and wait for port to open:  
    Serial.begin(9600);  
    // This delay gives the chance to wait for a Serial Monitor without  
blocking if none is found  
    delay(1500);  
}
```

```

// Defined in thingProperties.h
initProperties();

// Connect to Arduino IoT Cloud
ArduinoCloud.begin(ArduinoIoTPreferredConnection);

/*
    The following function allows you to obtain more information
    related to the state of network and IoT Cloud connection and
errors
    the higher number the more granular information you'll get.
    The default is 0 (only errors).
    Maximum is 4
*/
setDebugMessageLevel(2);
ArduinoCloud.printDebugInfo();

pinMode(rgbR, OUTPUT);
pinMode(rgbG, OUTPUT);
pinMode(rgbB, OUTPUT);
pinMode(led_pin1b, OUTPUT);

//PIR Setup
pinMode(led_pir, OUTPUT);
pinMode(pir_sensor_pin, INPUT);

//DHT11 Setup
pinMode(dht_sensor_pin, INPUT);
dht_sensor.begin();

lcd.begin(); // initialize the lcd
lcd.backlight();
}

void loop() {
    ArduinoCloud.update();

    val_pir = digitalRead(pir_sensor_pin);

```

```
if(val_pir == HIGH){
    digitalWrite(led_pir,HIGH);
    if(motion_thing == LOW){
        Serial.println("Motion Detected");
        motion_thing == HIGH;
    }
}else{
    digitalWrite(led_pir,LOW);
    if(motion_thing == HIGH){
        Serial.println("Motion Ended, no detection.");
        motion_thing == LOW;
    }
}

humidity = dht_sensor.readHumidity();
temp = dht_sensor.readTemperature();

if(isnan(humidity) || isnan(temp)){
    Serial.println("Failed to read from DHT11 Sensor.");
}
else{
    //DHT11 Information in the Serial Monitor
    Serial.print("Humidity: ");
    Serial.print(humidity);
    Serial.print("%");
    Serial.print(" | ");
    //Temperature Data
    Serial.print("Temperature: ");
    Serial.print(temp);
    Serial.println("°C");
    //LCD Information
    lcd.setCursor(0,0);
    lcd.print("Temp: ");
    lcd.print(temp);
    lcd.print((char)223);
    lcd.print("C");
    lcd.setCursor(0,1);
    lcd.print("Humidity: ");
```



```

    lcd.print(humidity);
    lcd.print("%");
}
delay(1250);

if(temp < 10){
    analogWrite(rgbR, HIGH);
    analogWrite(rgbG, HIGH);
    analogWrite(rgbB, LOW);
}else if(10 <= temp && temp <= 25){
    analogWrite(rgbR, LOW);
    analogWrite(rgbG, HIGH);
    analogWrite(rgbB, HIGH);
}
else{
    analogWrite(rgbR, HIGH);
    analogWrite(rgbG, LOW);
    analogWrite(rgbB, HIGH);
}
}

/*
    Since BlueLedSwitch is READ_WRITE variable, onBlueLedSwitchChange()
is
    executed every time a new value is received from IoT Cloud.
*/
void onBlueLedSwitchChange() {
    // Add your code here to act upon BlueLedSwitch change
    if(blue_led_switch == true){
        digitalWrite(led_pin1b, HIGH);
    }
    else{
        digitalWrite(led_pin1b, LOW);
    }
}

/*
    Since RedSwitch is READ_WRITE variable, onRedSwitchChange() is

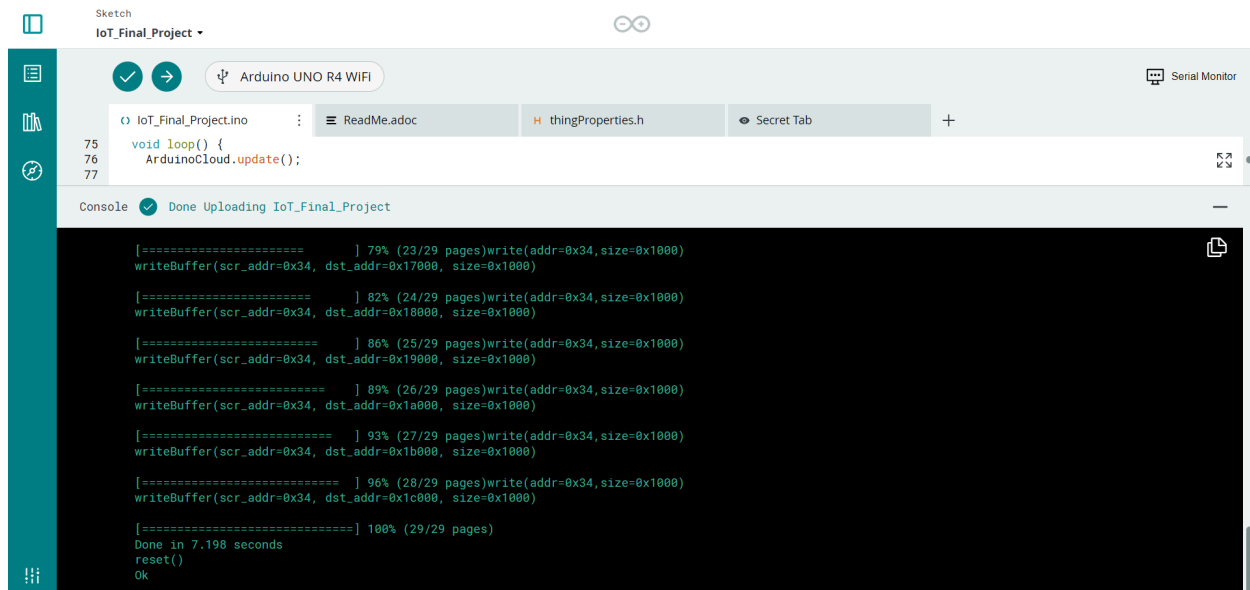
```

```

    executed every time a new value is received from IoT Cloud.
*/
void onRedSwitchChange() {
    int brightness = map(red_switch.getBrightness(), 0, 100, 0, 255);
    analogWrite(LED_PWM_PIN, brightness); // AnalogWrite for PWM
control
}

```

Verifying and Uploading the Code on the Arduino UNO R4 WiFi:



References

- <https://cloud.arduino.cc/>
- <https://docs.arduino.cc/arduino-cloud/cloud-interface/things/>
- <https://docs.arduino.cc/arduino-cloud/cloud-interface/dashboard-widgets/>
- <https://docs.arduino.cc/arduino-cloud/features/advanced-chart/>
- <https://docs.arduino.cc/arduino-cloud/iot-remote-app/getting-started/>
- <https://www.arduino.cc/reference/en/libraries/pir/>
- https://projecthub.arduino.cc/arduino_uno_guy/i2c-liquid-crystal-displays-5eb615
- <https://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/>
- <https://docs.arduino.cc/arduino-cloud/cloud-interface/variables/>
- <https://www.arduino.cc/reference/en/>