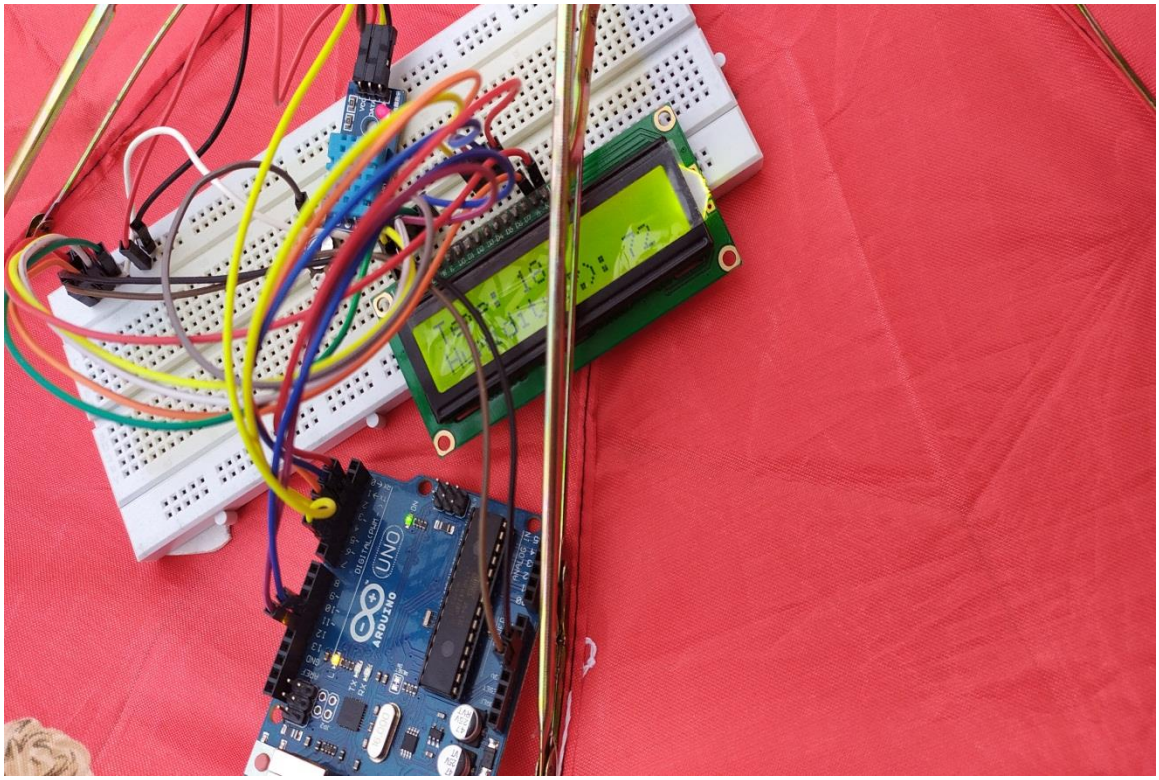
Smart Umbrella Using Arduino

[1st semester project, B.VOC(IOT)]

Introduction: Smart Umbrella is choose to improve existing umbrella. A humidity sensor senses, measures and regularly report the relative humidity in air it measures both humidity and air temperature as a percent in the ratio of actual humidity in the air to the highest amount of humidity air at that temperature can hold.

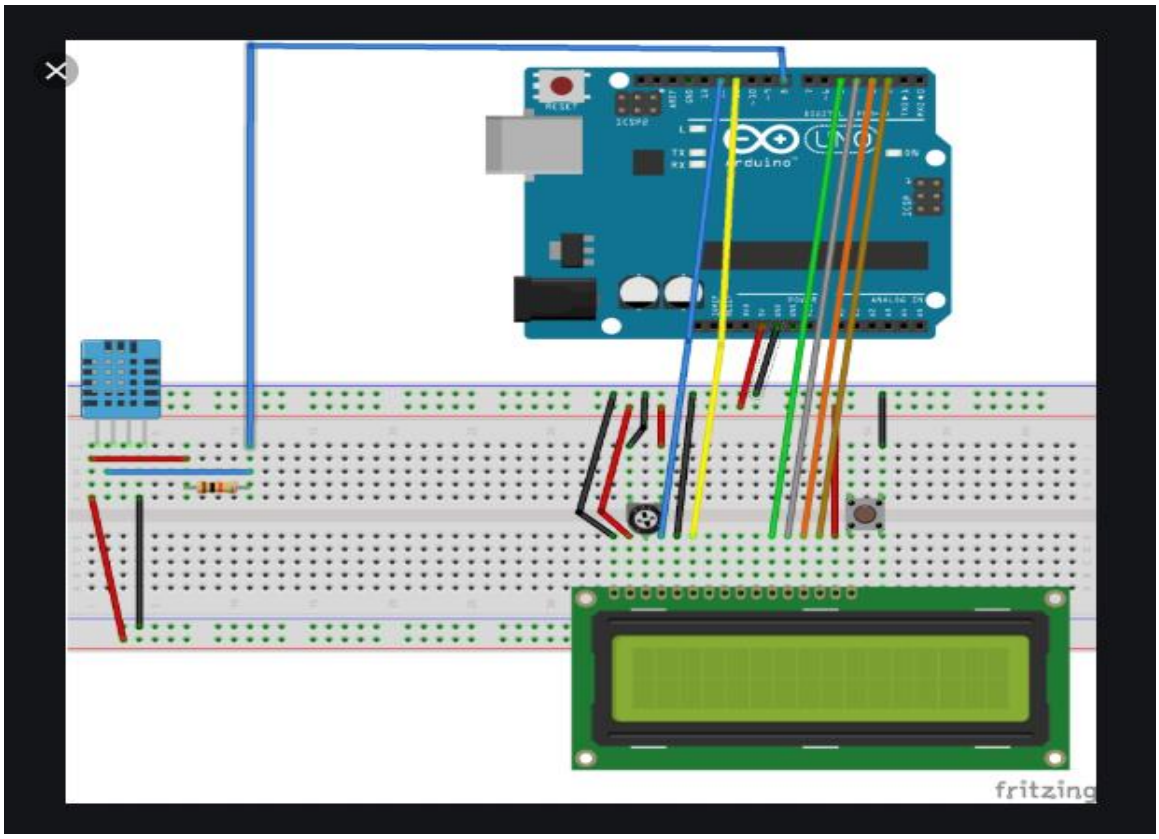
We can easily carry an umbrella with us so that we can see the temperature and humidity on it.



Components Required:

1. Arduino UNO
2. DHT11 sensor
3. LCD16*2 module
4. Breadboard
5. Potentiometer
6. Jumper wires

Circuit Diagram:



Programming Arduino For Smart Umbrella:

```
#include <LiquidCrystal.h>

#include <SimpleDHT.h>

//Declaring digital pin no 6 as the dht11 data pin

int pinDHT11 = 6;
SimpleDHT11 dht11;

//Declaring the lcd pins

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
// Don't forget to choose 9600 at the port screen

  Serial.begin(9600);
```

```
//Telling our lcd to start up

  lcd.begin(16, 2);


}

void loop() {

  //These serial codes are for getting readings on the port screen aswell as the LCD display, since they'll offer us a more detailed interface


  Serial.println("================================");
  Serial.println("DHT11 readings...");


  byte temperature = 0;
  byte humidity = 0;
  int err = SimpleDHTErrSuccess;

  //This bit will tell our Arduino what to do if there is some sort of an error at getting readings from our sensor
  if ((err = dht11.read(pinDHT11, &temperature, &humidity, NULL)) != SimpleDHTErrSuccess)
{
    Serial.print("No reading , err="); Serial.println(err);delay(1000);
    return;
  }

  Serial.print("Readings: ");
  Serial.print((int)temperature); Serial.print(" Celcius, ");
  Serial.print((int)humidity); Serial.println(" humidity%");

  //Telling our lcd to refresh itself every 0.75 seconds
  lcd.clear();

  //Choosing the first line and row
  lcd.setCursor(0,0);
  //Typing Temp: to the first line starting from the first row
  lcd.print("Temp: ");
  //Typing the temperature readings after "Temp: "
  lcd.print((int)temperature);
  //Choosing the second line and first row
  lcd.setCursor(0,1);
  //Typing Humidity(%): to the second line starting from the first row
  lcd.print("Humidity(%): ");
  //Typing the humidity readings after "Humidity(%): "
  lcd.print((int)humidity);
 delay(750);
}
```

Application:

1. Automation leads to better monitoring of device.
2. Efficient and save time.
3. Save money.
4. Better quality of life.

Submitted by-

Prerna Goyal and Nandini Varuns