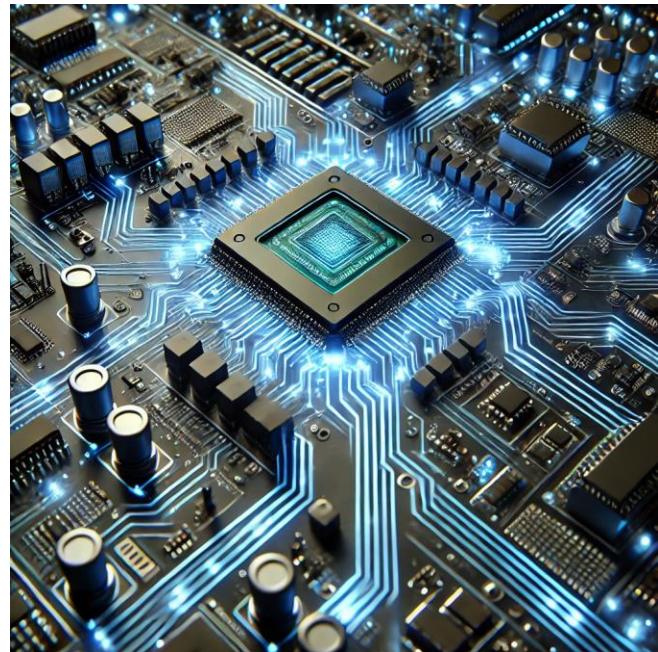


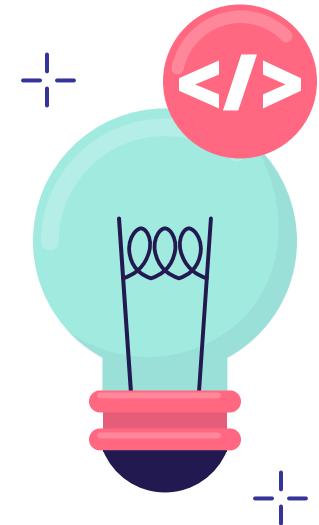
# Afficher la température et l'humidité mesurées par le capteur DHT11 connecté à Arduino sur l'afficheur LCD I2C

Présentée par :  
Elkhoulati Yahya



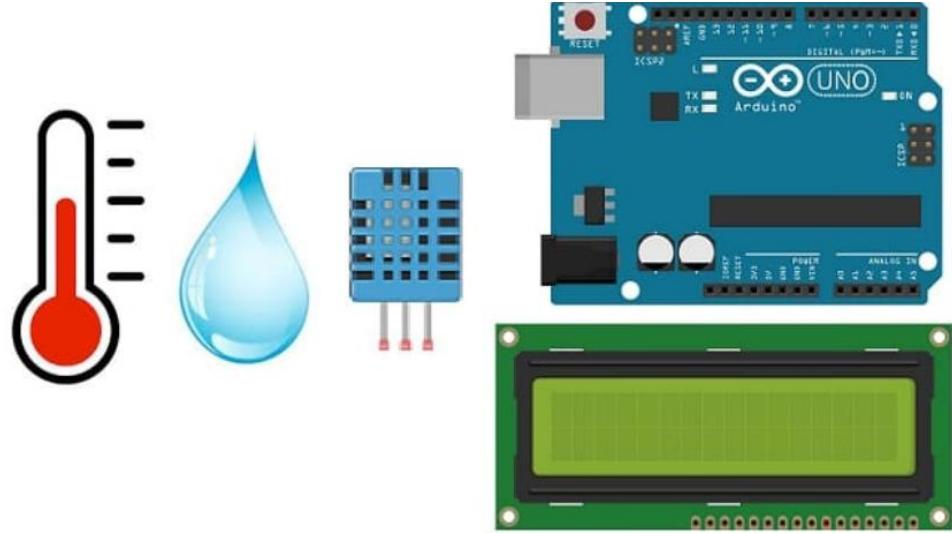
# Plan

- 01      Introduction et Objectifs**
- 02      Matériel et Composants Nécessaires**
- 03      Conception et Assemblage**
- 04      Programmation et Implémentation**
- 05      Résultats et Conclusion**



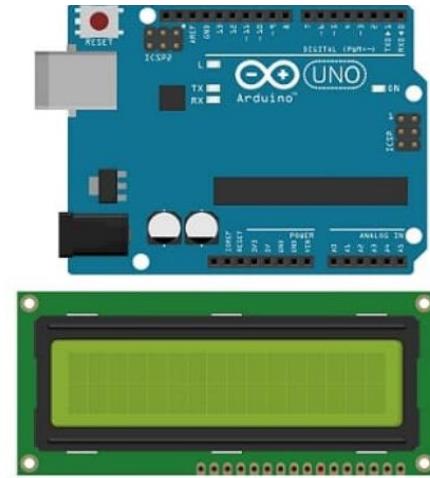
# 01

## Introduction et Objectifs



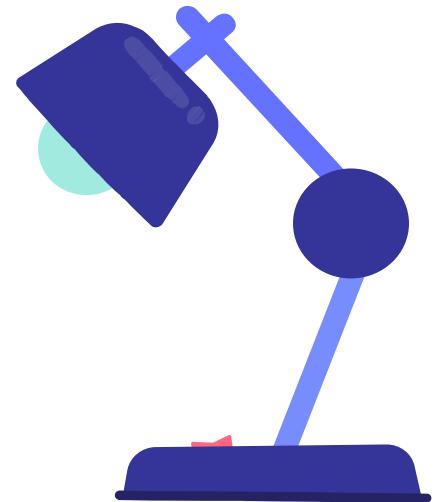
# Introduction :

Avec la surveillance des conditions environnementales devenant une nécessité dans le monde moderne, il devient critiquement crucial d'avoir des outils simples, abordables et épurés. Mesures de paramètres, à savoir la température et l'humidité, sont des éléments clés dans une gamme d'applications allant de la gestion des bâtiments, l'agriculture à l'essuie-glace et l'éclairage. Ce document décrit l'"Exécution d'un Monitor de Température et Humidité en Temps Réel" qu'est un système capable de mesurer ces paramètres et de les afficher en temps réel, offrant une solution abordable et facile à créer.



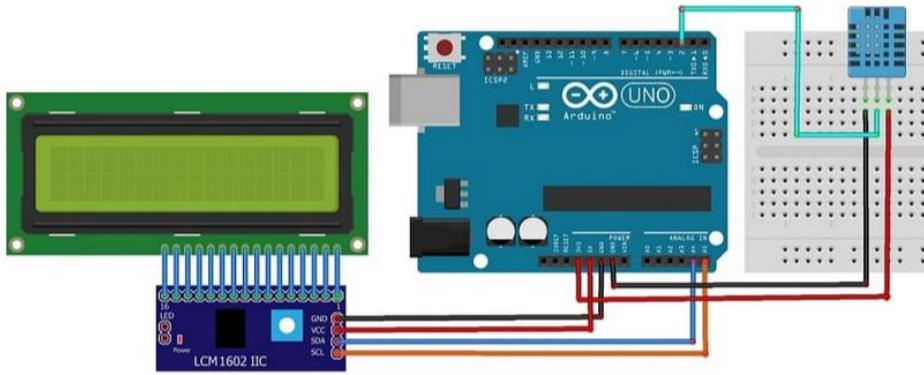
# Objectifs:

- 1. Mesurer la température et l'humidité** : Lire avec précision la température (en °C) et l'humidité relative (en %) grâce au capteur DHT11.
- 2. Afficher les données en temps réel** : Fournir une visualisation claire et directe des mesures sur un écran LCD.
- 3. Suivi à distance** : Permettre de lire les informations également sur un ordinateur via le moniteur série.
- 4. Fiabilité et gestion des erreurs** : Déetecter les anomalies ou erreurs de lecture du capteur et informer l'utilisateur à travers des messages d'alerte.
- 5. Design Rythmie vs Accessibilite**: Pour ce qui est de l'accessibilite, on choisit d'avoir un materiel a bon mache et facile a assembler pour que le réseau de capteur soit une solution pratique et economique.
- 6. Portalite d'extension** : dans le cas ou on veut s'étendre et ajouter quelques fonctionnalite en plus sachant qu'elle peut aider a etre automatise ou innove en IOT par example.



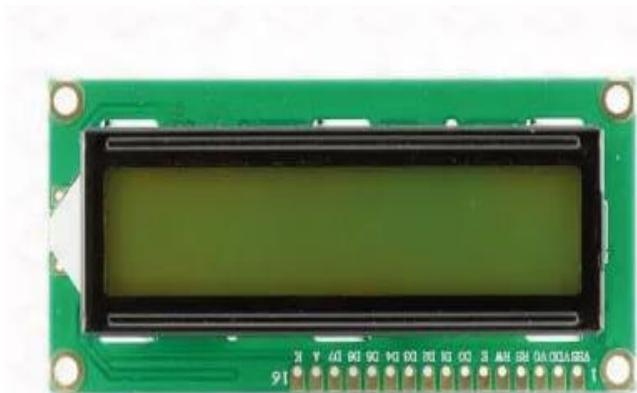
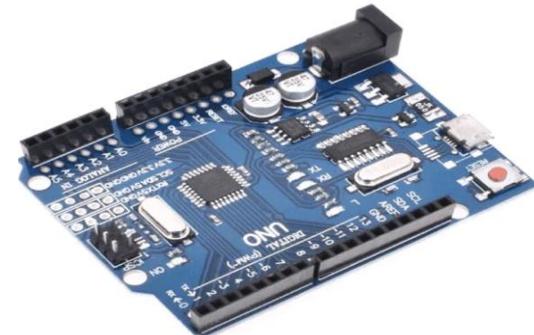
# 02

# Matériel et Composants Nécessaires



# Les composants

- Carte Arduino Uno :
  - **Rôle** : La carte Arduino servira de cerveau pour contrôler et interagir avec les différents composants.
- Afficheur LCD I2C :
  - **Rôle** : Un écran LCD qui simplifie la connexion à l'Arduino en utilisant le protocole I2C. Il comprend un contrôleur intégré qui réduit le nombre de broches nécessaires pour le connecter à l'Arduino.
    - Il est utilisé pour afficher la température et l'humidité mesurées par le capteur DHT11.



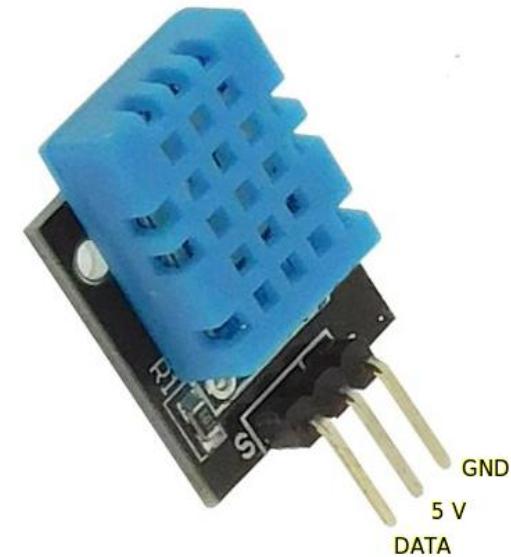
## • Le capteur DHT11 :

=>Le capteur DHT11 est un capteur d'humidité et de température relativement basique, mais populaire et facile à utiliser dans des projets électroniques et d'IoT (Internet des objets). Ce capteur est abordable et peut être intégré à des microcontrôleurs tels que l'Arduino pour mesurer à la fois l'humidité relative et la température ambiante.

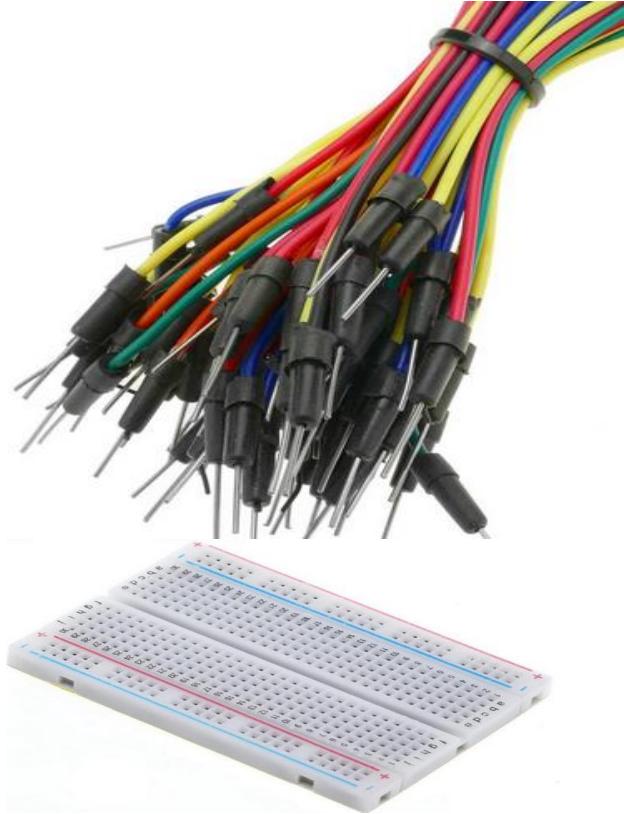
=>Caractéristiques:

- 1. Mesure de l'humidité**
- 2. Mesure de la température**
- 3. Sortie numérique**

-**Rôle** : Mesurer la température et l'humidité de l'environnement.

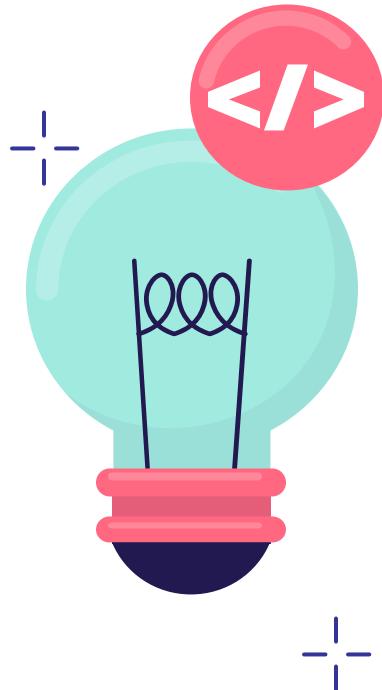


- **Câbles de connexion:**  
-**Rôle** : Relier physiquement les composants entre eux.
- **Breadboard (plaqué d'essai):**  
-**Rôle** : Facilite les connexions temporaires entre les composants électroniques sans soudure.



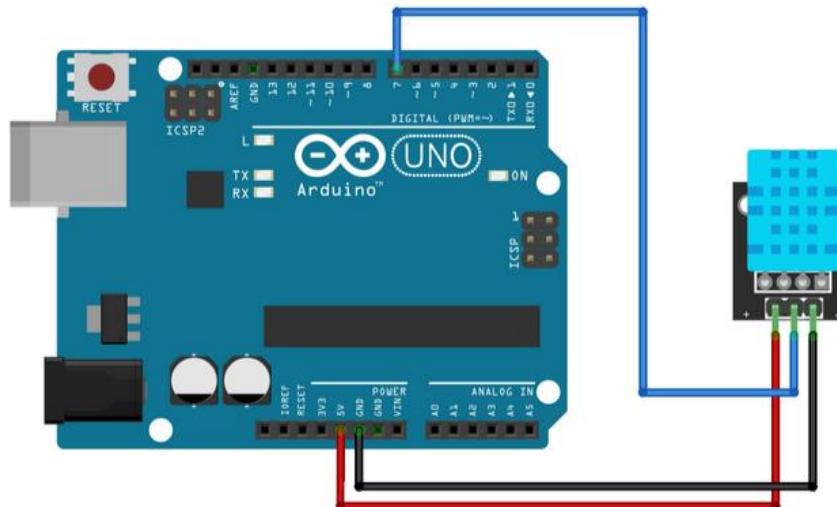
03

# Conception et Assemblage

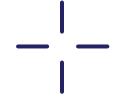


# Schéma de câblage de la carte Arduino avec le capteur DHT11

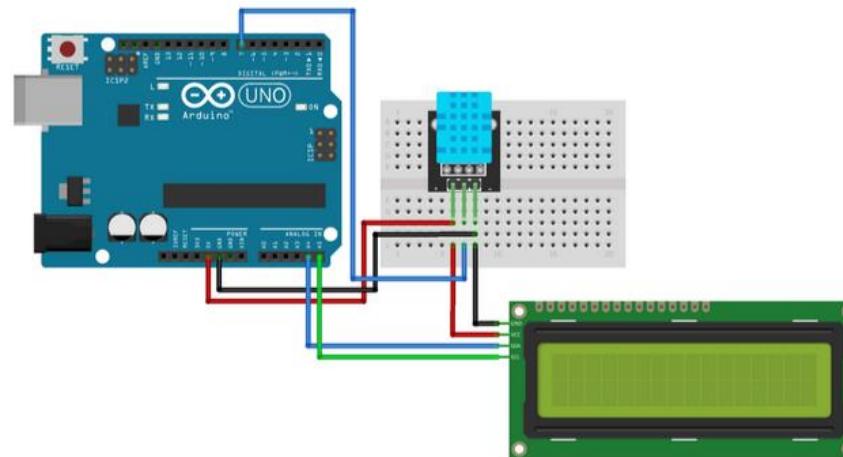
=>Voici un schéma de câblage pour connecter la carte Arduino avec le capteur DHT11:



# Schéma de câblage de la carte Arduino avec le capteur DHT11 et l'afficheur LCD I2C



=>Voici un schéma de câblage pour connecter la carte Arduino avec le capteur DHT11 et l'afficheur LCD I2C :



# Étapes d'Assemblage

## .Pour le Capteur DHT11 :

- Connecter la broche VCC du DHT11 à 3.3V de l'Arduino UNO
- Connecter la broche DATA du DHT11 à la broche Pin 2 de l'Arduino UNO.
- Connecter la broche GND du DHT11 à GND de l'Arduino.

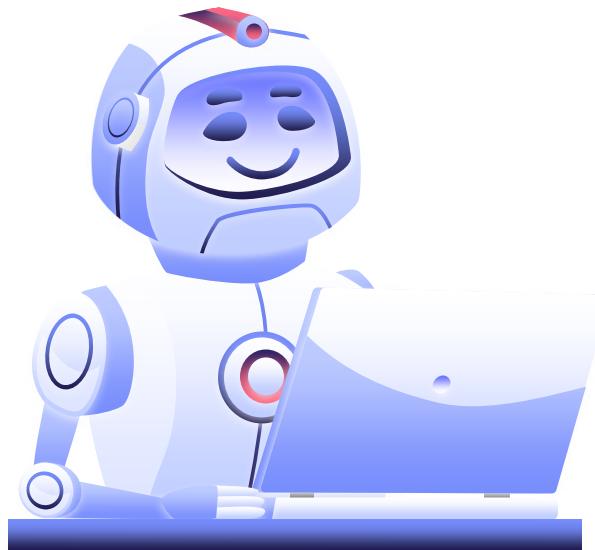
## .Pour l'Afficheur LCD I2C :

- Connecter la broche VCC de l'afficheur à 5V de l'Arduino.
- Connecter la broche GND de l'afficheur à GND de l'Arduino.
- Connecter la broche SDA de l'afficheur à la broche A4 (Analogique 4) de l'Arduino.
- Connecter la broche SCL de l'afficheur à la broche A5 (Analogique 5) de l'Arduino.

.Assurez-vous que les broches SDA et SCL de votre afficheur LCD I2C correspondent aux bonnes broches de votre modèle. Il est également crucial de respecter la polarité (connecter correctement VCC à 5V et GND à GND) pour éviter d'endommager les composants.

.Une fois que vous avez connecté correctement les composants, vous pouvez utiliser le code Arduino approprié pour lire les données du capteur DHT11 et les afficher sur l'afficheur LCD I2C, comme mentionné précédemment.





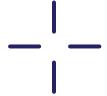
04

# Programmation et Implémentation



Elkhoulati Yahya

# Programmation de la carte Arduino UNO pour afficher la température et l'humidité mesurées par le capteur DHT11 sur l'afficheur LCD I2C



Voici le code Arduino pour lire la température et l'humidité à partir du capteur DHT11 et afficher ces données sur un afficheur LCD I2C :

Pour ce code, vous aurez besoin des bibliothèques LiquidCrystal\_I2C pour l'afficheur LCD I2C et DHT pour le capteur DHT11. Assurez-vous de les installer via le Gestionnaire de bibliothèques de l'IDE Arduino.



### dht11\_project.ino

```
1 #include "DHT.h"
2 #include <Wire.h>          // Bibliothèque pour la communication I2C
3 #include <LiquidCrystal_I2C.h> // Bibliothèque pour l'écran LCD i2C
4
5 // Configuration du capteur DHT11
6 #define DHTPIN 2            // Pin connectée à la broche DATA du capteur
7 #define DHTTYPE DHT11        // Type de capteur : DHT11
8
9 DHT dht(DHTPIN, DHTTYPE); // Initialisation de l'objet DHT
10
11 // Configuration de l'écran LCD (adresse i2C, colonnes, lignes)
12 LiquidCrystal_I2C lcd(0x27, 16, 2);
13 void setup(){
14     Serial.begin(9600);      // Initialisation de la communication série
15     dht.begin();             // Initialisation du capteur DHT11
16     lcd.init();              // Initialisation de l'écran LCD
17     lcd.backlight();         // Activer le rétroéclairage de l'écran
18     lcd.setCursor(0, 0);     // Positionnement du curseur sur la première ligne
19     lcd.print("Initialisation");
20     delay(2000);             // Pause pour afficher le message d'accueil
21     lcd.clear();             // Efface l'écran avant le début des mesures
22 }
```

```
23 void loop(){
24     // Lecture des données du capteur
25     float temp = dht.readTemperature(); // Lecture de la température en Celsius
26     float hum = dht.readHumidity();     // Lecture de l'humidité en %
27     // Vérification des erreurs
28     if(isnan(temp) || isnan(hum)){
29         Serial.println("Erreur : Données invalides du capteur !");
30         lcd.setCursor(0, 0);
31         lcd.print("Erreur capteur !");
32         lcd.setCursor(0, 1);
33         lcd.print("Vérifiez liaison");
34     }else{
35         // Affichage des mesures sur le Moniteur Série
36         Serial.println("Température : " + String(temp) + " °C");
37         Serial.println("Humidité : " + String(hum) + " %");
38         // Affichage des mesures sur l'écran LCD
39         lcd.setCursor(0, 0); // Position sur la première ligne
40         lcd.print("Temp: ");
41         lcd.print(temp);
42         lcd.print(" °C");
43         lcd.setCursor(0, 1); // Position sur la deuxième ligne
44         lcd.print("Hum: ");
45         lcd.print(hum);
46         lcd.print(" %");
47     }
48     // Pause de 10 secondes avant la prochaine lecture
49     delay(10000);
50 }
```



# Explications

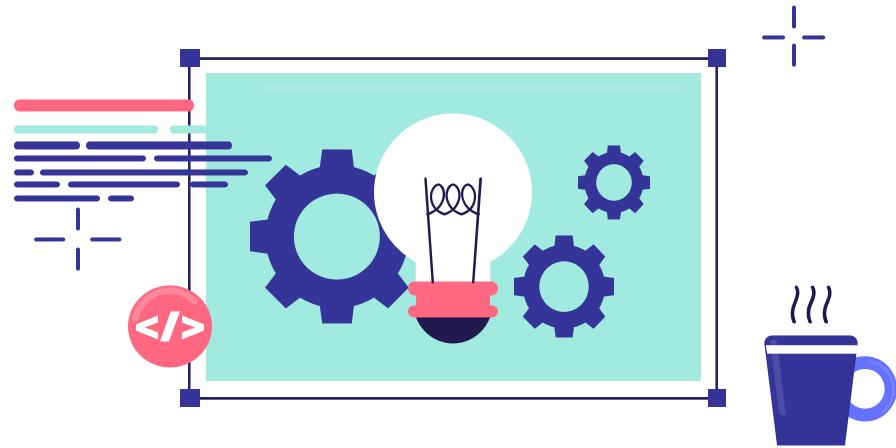
Ce code initialise le capteur DHT11 et l'afficheur LCD, puis lit périodiquement la température et l'humidité du capteur. Il affiche ensuite ces valeurs sur l'afficheur LCD I2C.

Téléversez ce code sur votre Arduino et vous devriez voir les lectures de température et d'humidité s'afficher sur votre afficheur LCD I2C.



# 05

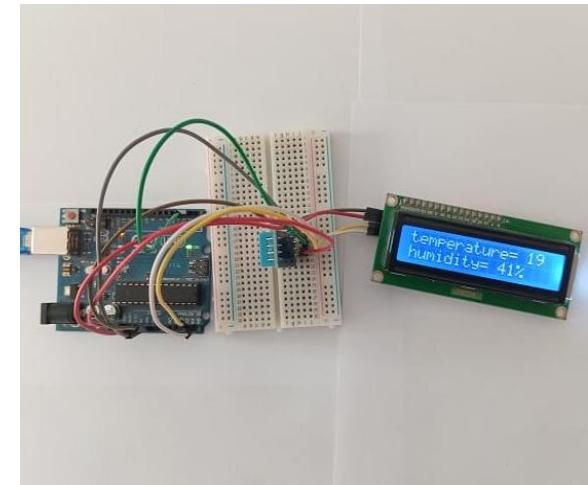
# Résultats et Conclusion



# Résultats

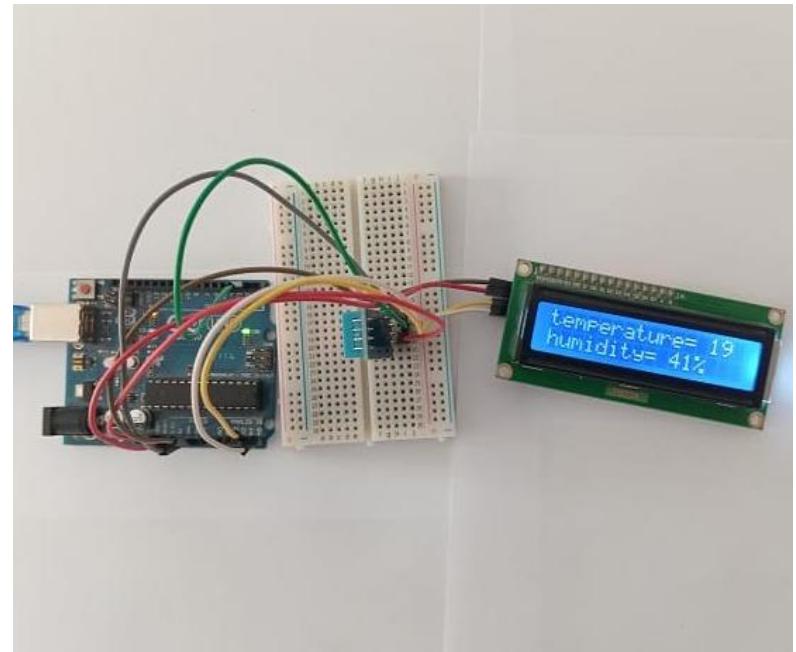
Le moniteur de température et d'humidité fonctionne parfaitement et remplit les objectifs fixés. Voici les principaux résultats :

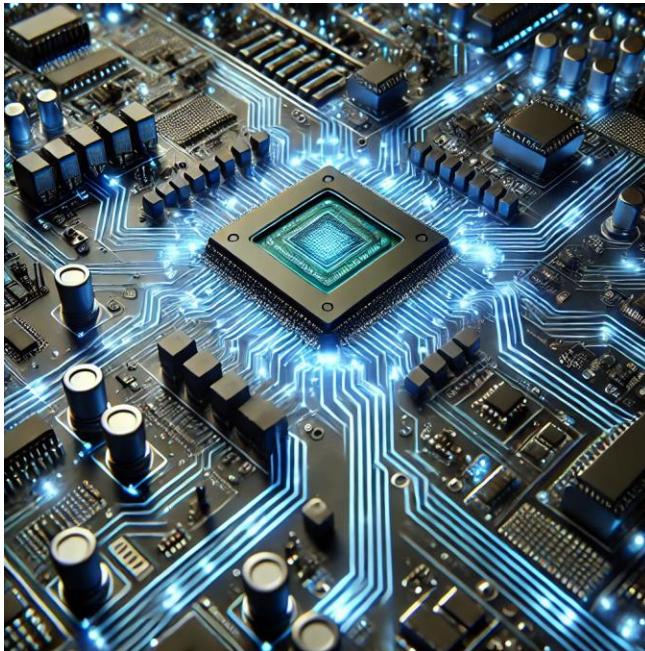
- 1.**Les valeurs de température et d'humidité sont affichées en temps réel sur l'écran LCD I2C, avec une excellente lisibilité.
- 2.**En cas d'erreur, comme un capteur déconnecté ou des données incohérentes, le système affiche un message clair sur l'écran et active le buzzer pour alerter l'utilisateur.
- 3.**Les tests de précision montrent que les données fournies par le capteur DHT11 sont globalement fiables pour des usages simples, même si elles restent perfectibles pour des applications nécessitant une grande précision.
- 4.**Le système s'est montré stable et performant après plusieurs heures d'utilisation continue, sans interruption ou comportement imprévu.



# Conclusion

Ce projet est un succès ! Il a permis de concevoir un outil simple mais efficace pour surveiller la température et l'humidité en temps réel. Que ce soit pour un usage personnel, éducatif ou comme base pour un projet IoT plus avancé, ce moniteur est pratique et fonctionnel.





MERCI POUR  
VOTRE  
ATTENTION



Elkhoulati Yahya