

# IoT-Based Temperature and Humidity Monitoring System

An IoT-based temperature and humidity monitoring system using Arduino, DHT11 sensor, and Python for data analysis. This beginner-friendly project is perfect for students looking to explore the Internet of Things and data visualization. This repository is based on <https://siliconwit.com/iot/temperature-and-humidity-monitoring-system>

## Project Schematic

This folder contains a Fritzing files that represent how the various components are connected to the arduino. The project currently implements an arduino uno but an arduino mega could be used if more sensors were to be included to provide more functionalities.

- *color coding for fritzing files*: black: represents connection to individual arduino pins. green: represents connection to ground. red: represents connection to voltage supply (5V), if a connection is made to the 3.3V source be sure to state that explicitly in the documentation. Maintain this format when adding more components to make it easy to understand.

## PCB-design

This folder contains PCB designs for the circuit connection

## Arduino code

This folder contains Arduino code for the project. The arduino connection contains : Arduino Uno, Liquid Crystal Display(LCD) , DHT Sensor , buzzer , RGB LED and current limiting resistors

- The arduino code programs the DHT sensor to read the temperature and humidity values from the environment and display the data on the serial monitor and on the LCD.
- The RGB LED and the buzzer are for alerts when the temperature or humidity values reach a certain set values

## Areas of Improvement

The current implementation of the IoT temperature and humidity monitoring system serves as a foundation for a beginner-friendly project. However, there are several areas where contributors can improve the project, making it more versatile and feature-rich:

- *Wireless communication:* Integrate a Wi-Fi or Bluetooth module (e.g., ESP8266, HC-05) to send data wirelessly from the Arduino to a remote device or server. This allows for real-time monitoring of temperature and humidity data on a smartphone or computer, even from a distance.
- *Web dashboard:* Develop a web dashboard to display temperature and humidity data in a user-friendly manner. This can be achieved using web technologies such as HTML, CSS, JavaScript, and web frameworks like Flask or Django.
- *Alerts and notifications:* Implement a system to send alerts or notifications via email, SMS, or other messaging platforms when certain thresholds are met, such as when the temperature or humidity exceeds a predefined limit.
- *Data storage:* Store the collected data in a database (e.g., SQLite, PostgreSQL, InfluxDB) for long-term storage and analysis. This will allow for better historical data analysis and trend identification.
- *Advanced data analysis:* Utilize machine learning techniques or statistical methods to predict future temperature and humidity levels, identify trends, or detect anomalies in the data.
- *Improved hardware design:* Design and build a custom PCB or enclosure for the project, making it more robust and suitable for deployment in various environments.
- *Battery-powered operation:* Integrate a battery power source and power management system to make the project portable and usable in locations without a direct power supply.
- *Additional sensors:* Incorporate other environmental sensors, such as air quality sensors, light sensors, or sound sensors, to create a comprehensive environmental monitoring system.

By working on these areas of improvement, contributors can help enhance the project's functionality and expand its potential applications, making it more valuable and appealing to a broader audience.

## Improvement

- We improved the original program to the current 'PROJECT.py' and 'proj\_plot.py'
- For the arduino code we added a LCD and HC-06 bluetooth module

