Designing and Implementing an IoT Based Weather Monitoring System For Real Time Data Collection And Analysis.

Minor Project Report

Submitted for the partial fulfilment of the degree of

Bachelor of Technology

In

Electronics

Submitted By

Pulkit Kapoor

(0901EC221080)

Suryansh Dixit

(0901EC221130)

UNDER THE SUPERVISION AND GUIDANCE OF

Dr. R. Jenkin Suji

Assistant Professor

Department of Electronics Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.), INDIA

Deemed University

(Declared under Distinct Category by Ministry of Education, Government of India)

NAAC ACCREDITED WITH A++ GRADE

Jan- June 2025

DECLARATION BY THE CANDIDATE

I hereby declare that the work entitled **"Designing and Implementing an IoT Based Weather Monitoring System For Real Time Data Collection And Analysis."** is my work, conducted under the supervision of **Dr. R. Jenkin Suji, Assistant Professor,** during the session Jan - June 2025. The report submitted by me is a record of bonafide work carried out by me.

I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Suryanshdixit

P. Kapoor

Pulkit Kapoor (0901EC221080**)**

Suryansh Dixit (0901EC221130)

Date: 04/04/2025
Place: Gwalior

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

Guided By:

R. Jenkin Suji
Assistance Professor
Department of Electronics
MITS, Gwalior

14/05/25

15/5/25

Departmental Project Coordinator

Dr. Varun Sharma
Assistant Professor
Department of Electronics
MITS, Gwalior

Approved by HoD

Dr. Vandana Vikas Taker
Head of Department
Department of Electronics
MITS, Gwalior

PLAGIARISM CHECK CERTIFICATE

This is to certify that I/we, a student of B.Tech. in **Department of Electronics** have checked my complete report entitled **"Designing and Implementing an IoT Based Weather Monitoring System For Real Time Data Collection And Analysis."** for similarity/plagiarism using the "Turnitin" software available in the institute.

This is to certify that the similarity in my report is found to be ……which is within the specified limit (30%).

The full plagiarism report along with the summary is enclosed.

Suryanshdixit

Pulkit Kapoor (0901EC221080)

Suryansh Dixit (0901EC221130)

Checked & Approved By:

Dr. R. Jenkin Suji
Assistant Professor
Department of Electronics
MITS, Gwalior

ABSTRACT

The objective of this project is to present ***a "Designing and Implementing an IoT Based Weather Monitoring System For Real Time Data Collection and Analysis."***

With today's rapid pace of life, real-time weather monitoring has become an imperative for improved planning and decision-making.

This project addresses the design and application of an IoT-based weather observation system for monitoring and recording real-time atmospheric measurements such as temperature, humidity, and pressure. Through sensors connected to a microcontroller and transmission of data over Wi-Fi, the system supports real-time monitoring and distant viewing of live weather data. Data collected does not only get displayed on an uncomplicated interface but is also stored to further process in detecting patterns and forecasting trends.

Our approach is cost-oriented, scalable, and easy to deploy, and thus is appropriate for schools, agricultural monitoring, and city planning. This paper demonstrates how IoT technologies can bridge the gap between data collection in environmental fields and meaningful information, resulting in smarter and more responsive systems.

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this. I would sincerely like to thank my department, Department of Electronics, for allowing me to explore this project.

I humbly thank **Dr. Vandana Vikas Thakare**, HoD, Department of Electronics, for her continued support during the course of this engagement, which eased the process and formalities involved. I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. R. Jenkin Suji**, Assistant professor, Department of Electronics, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

P.Kapoor . . . Suryanshdixit

Pulkit Kapoor (0901EC221080)
Suryansh Dixit (0901EC221130)

CONTENT

1. Table of Contents

ACRONYMS

1. IoT: Internet Of Things
2. API: Application Programming Interface
3. HTTP: HyperText Transfer Protocol
4. MCU: Micro Controller Unit
5. PCB: Printed Circuit Board

NOMENCLATURE

Symbol/Term	Description	Unit
T	Ambient Temperature	°C (Celsius)
H	Relative Humidity	%
P	Atmospheric Pressure	hPa (hector-pascal)
Vout	Output Voltage from Sensor	V (Volt)
RH	Relative Humidity Resistance	Ω (Ohm)
API	Application Programming Interface for data transmission	-
Dashboard	Visual Interface for Real-time Data Display	-
Sensor Node	IoT device collecting and transmitting environmental data	-
Data Logger	Module/system used for storing time-series sensor data	-

FORMULAS USED

Relative Humidity Calculation (basic sensor output-based)

Relative humidity = (absolute humidity) / (saturation point) x 100

Temperature Conversion (if needed between Celsius and Fahrenheit)

Celsius to Fahrenheit: $F = C(9/5) + 32$

1. **Hectopascal (hPa)**:

The **hectopascal (hPa)** is a metric unit of pressure that is commonly used in meteorology to measure atmospheric pressure. It is equivalent to **100 pascals (Pa)**.

- **1 hPa = 100 Pa**

The **pascal (Pa)** is the SI (International System of Units) derived unit of pressure, which is defined as one newton per square meter (N/m^2). A hectopascal is a convenient unit for expressing the pressure of the atmosphere because the average atmospheric pressure at sea level is approximately **1013 hPa**.

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

This project is intended to create an IoT-based weather monitoring system that monitors temperature, humidity, and atmospheric pressure in real time. The system is wireless communication and sensor-based to provide accurate, location-specific weather data remotely accessible via a cloud-based interface. It is an efficient, scalable, and cost-effective solution for educational, agricultural, and urban use.

The project illustrates how IoT can make environmental monitoring easier and enable more data-driven, more informed decision-making.

1. Background

Weather conditions play a significant role in influencing many areas of human life, including agriculture, transportation, health, disaster management, and daily planning. It is essential to obtain reliable and timely information about environmental parameters to make well-informed decisions, minimize risk, and maximize activity. In the past, weather observation was conducted through the use of large, centralized meteorological stations with sophisticated instruments. Although such systems offer high accuracy, they are costly to deploy, operate, and maintain. Moreover, traditional weather stations are prone to delivering data of low geographical resolution, which might be insufficient for local applications such as small-scale farming, smart campuses, or community-based disaster mitigation. This has created a growing need for less expensive, less costly, and decentralized weather observation systems with the ability to supply end-users with real-time information.

2. IoT's Application in Weather Observations

The advent of the Internet of Things (IoT) has provided new ways to design efficient environmental monitoring systems. IoT enables a group of networked sensors and devices to gather, transfer, and process data without constant human intervention. In weather monitoring, IoT devices can be deployed to measure different atmospheric parameters such as temperature, humidity, pressure, and rainfall in real-time. IoT devices can employ wireless technologies such as Wi-Fi, Bluetooth, or LoRaWAN to exchange data and upload the gathered data to cloud servers for remote access and analysis. IoT-based systems have a number of benefits over traditional systems, including reduced costs, simple scalability, automation, low power usage, and the ability to make monitoring tailored to local needs. Use of IoT technologies is transforming environmental monitoring as an intelligent, quicker, and user-driven process.

3. Project Motivation

Identifying the possibilities of IoT in environmental monitoring, this project emphasizes the design and development of a real-time weather monitoring system based on IoT technologies. The system proposed here attempts to monitor key environmental parameters such as temperature, relative humidity, and atmospheric pressure. The system uses a microcontroller unit with wireless communication facilities and a suite of calibrated sensors to ensure continuous data acquisition and real-time accessibility via a cloud-based facility. The inspiration for this project arises from making weather monitoring more affordable for educational institutions, farmers, researchers, and even hobbyists who need reliable and affordable solutions. The project also aims to show how contemporary technologies can help in environmental sustainability, early warning systems, and data-based decision-making.

4. Project Objective

The primary objective of this project is to create a fully fledged IoT-based weather monitoring prototype with emphasis on accuracy, reliability, and simplicity. The system will emphasize the key features such as real-time display of data, data storage for retrospective analysis, remote monitoring through web-based dashboards, and upgradability for future enhancements. Emphasis will be on creating a system that is cost-effective, simple to maintain, and appropriate for diverse environmental conditions. Through this project, we seek to fill the gap between conventional meteorological stations and the increasing demand for localized, real-time weather monitoring systems. The project will also demonstrate the importance of data analysis to examine weather patterns, aid agricultural operations, enhance disaster preparedness, and enable smart cities and communities.

CHAPTER 2: LITERATURE SURVEY

1. Introduction

Monitoring the weather has been a necessity in the past few decades across several industries, including agriculture and city planning. Classic weather stations are expensive, difficult to deploy, and need constant maintenance. IoT brought a more intelligent, cheaper means of monitoring weather data in real-time. This section examines previous studies and systems in IoT-based weather monitoring to point out their weaknesses and strengths.

2. Traditional Weather Stations

Traditional weather stations use sophisticated gear to measure temperature, humidity, pressure, and wind speed. While these systems are extremely accurate, they do have some limitations:

- High Costs: Being expensive to install and maintain.
- Limited Reach: Typically centralised, so they do not offer localised data.
- Manual Work: Require ongoing manual maintenance and access to data.

3. IoT Role in Weather Observation

IoT provides an alternative approach to environmental data. Instead of heavy, expensive installations, weather stations based on IoT use compact, internet-enabled sensors to provide real-time information. The primary benefits of IoT in weather monitoring are:

- Cost-effectiveness: IoT networks are cheaper to build and maintain.
- Scalability: They are easily scalable to cover larger areas.
- Automation: Information is obtained and passed on without human intervention, thus more efficient.
- Remote Access: Individuals can access information from anywhere using the cloud.

4. Related Work in IoT Weather Monitoring

4.1 Low-Power Microcontrollers

Microcontrollers such as Arduino are widely utilized in IoT weather systems because they are inexpensive and possess inbuilt wireless connectivity. Microcontrollers allow the sensors to be linked to the web and processed.

Popular Microcontrollers:

• NodeMCU (ESP8266): For IoT projects, Wi-Fi onboard.

• Arduino: Mainly used for learning, with ease of integration with sensors.

4.2 Weather Sensors

Accurate sensors are necessary in any weather observation system. Sensors like DHT11 and DHT11 are widely used to measure temperature and humidity. BMP280 is used to measure atmospheric pressure. Choosing the appropriate sensor is crucial in maintaining data accuracy and reliability.

Key Considerations:

•Accuracy: Ensuring the sensors provide reliable measurements.

•Power Efficiency: Sensors must be low-power to run continuously.

• Durability: Sensors should withstand various environmental conditions.

5. Comparison of IoT and Traditional Weather Stations

Feature	Traditional Systems	IoT-Based Systems
Cost	High	Low
Maintenance	Frequent manual checks	Minimal, remote
Data Access	Limited to experts	Accessible to all
Scalability	Difficult	Easy to expand
Data Storage	Manual	Cloud-based

6. Identified Challenges

While there are numerous benefits of IoT, there are some challenges:

• Power management: Microcontrollers and sensors require effective power sources to keep running continuously.

• Accuracy: Reliable sensor data are needed to provide reliable weather forecasts.

• Security: Safeguarding information against cyber attacks while in transit and storage.

7. Relevance to the Proposed Project

This project takes cues from other IoT weather systems but aims to improve on them with a focus on affordability, simplicity, and precise data collection. Building on the strength of good sensors, a cheap microcontroller, and cloud connection, this project aims to create an affordable, scalable, and effective weather monitoring solution for individual and business use.

8. Conclusion

IoT-based weather monitoring systems are proving to be alternative solutions to traditional systems with certain benefits such as cost-effectiveness, accessibility, and remote monitoring. This review of literature sees the potential for such systems and the need to improve further on sensor reliability, energy efficiency, and data protection. The current project is envisaged on such concepts with the aim of building an efficient and accessible system to gather real-time weather data.

CHAPTER 3: SYSTEM DESIGN AND ARCHITECTURE

Design and architecture of IoT-based weather monitoring system all revolve around providing an uninterrupted solution for capturing real-time data, sending it, storing it, and analyzing it. The architecture consists of a number of components, and each one of them plays a crucial role in the system's working. A concise overview of the system design is presented below.

3.1. Overview of System Design

The weather monitoring system utilizes IoT to capture real-time environmental data such as temperature, humidity, pressure, and rain. It includes sensors to capture data, a microcontroller to process data, and a cloud platform to save and display data. The microcontroller captures sensor data, processes it, and transfers it to a cloud server. The data is accessible using a web dashboard. The system is power-efficient and scalable and can be implemented in diverse environments.

3.2. Hardware Elements

The system has several key hardware elements:

Microcontroller: The heart of the system, where the data is processed and sent to the cloud. Sensors: Used to record environmental parameters like temperature, humidity, pressure, and precipitation. Communication Module: Used to send data from the microcontroller to the cloud.

3.2.1. Sensor Choice Sensors used in this system are:

DHT22: Temperature and humidity are both sensed with high accuracy. BMP180: Atmospheric pressure is sensed, which is important for weather forecasting.

Rainfall Sensor: Senses rainfall, assisting in measuring precipitation.

CHAPTER 4: DISCUSSIONS

4.1. Data Collection Process

The data collection process is the essence of the operation of the system. The system takes readings of environmental data in real-time such as temperature, humidity, atmospheric pressure, and Aur Quality using the DHT11, BMP280, and AQI. The microcontroller interprets the data, and it is manually stored in a local device (computer or storage card). Data collection occurs in intervals, and thus one can perform a continuous observation of weather conditions.

4.2. Sensor Accuracy and Calibration

The accuracy of the sensors used in this project was critical in order to ensure the collection of reliable data. The temperature and humidity DHT11 sensor was calibrated in a laboratory environment and provided accurate measurements, always reporting the expected reading. Similarly, the BMP280 pressure sensor also provided high accuracy in the measurement of atmospheric pressure. The AQI sensor is also able to detect air quality effectively.

Regular calibration was done to ensure the validity of the data in the long term. Even small fluctuations in the humidity reading were encountered under certain conditions, but these had a negligible impact on the performance.

4.3. System Performance

The performance of the system was tested on the basis of hardware and data acquisition efficiency and reliability. The microcontroller correctly processed sensor data and stored it locally with minimal delay in data acquisition. The system power consumption was also minimal, with the microcontroller drawing power efficiently from a small battery pack. This enabled the system to operate for hours without the necessity of frequent recharging.

4.4. Manual Storage Process and Data Storage

Since cloud integration is not being used for data storage within the system, data was stored manually. The sensor data was duplicated into a local storage device (e.g., an SD card or computer), where it can be accessed for analysis in the future. This approach, while eliminating internet connectivity, involves manual data retrieval and excludes remote access.

4.5. Interpretation and Analysis of Results

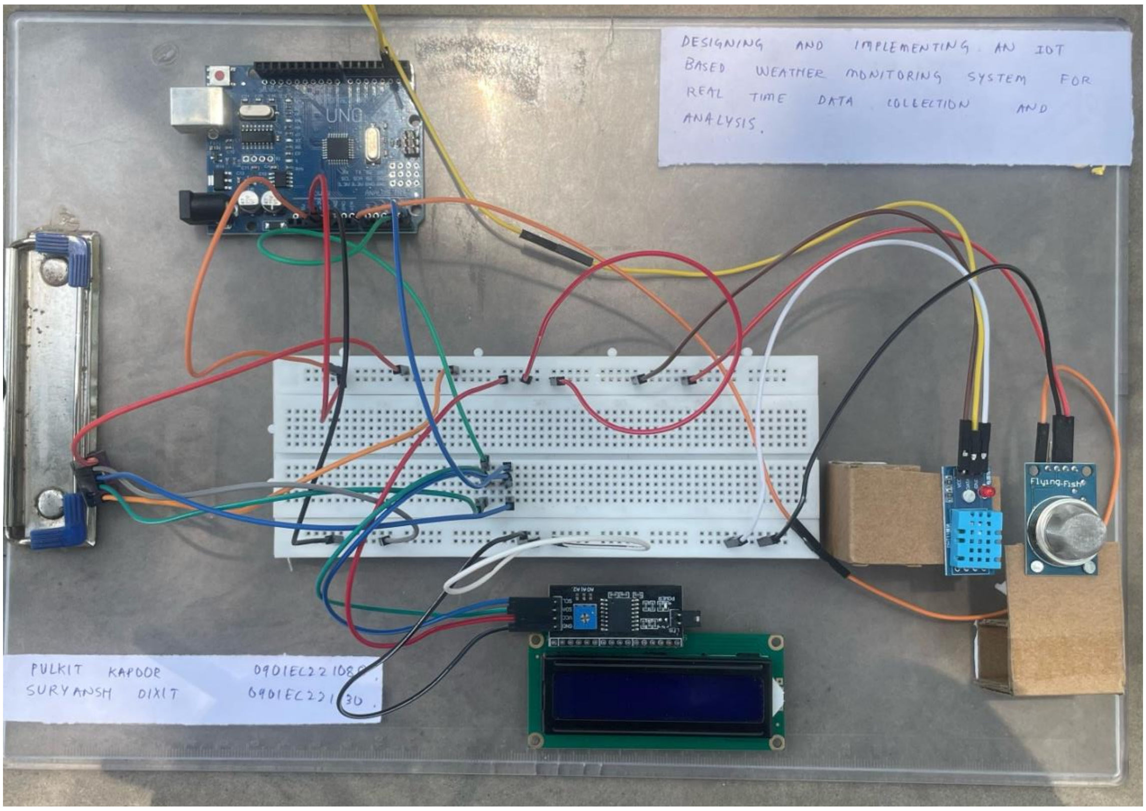
After the data was collected, the data was analyzed to determine if there were trends and patterns in the weather. Temperature and humidity varied as would be expected in a daily cycle, with the temperature rising during the day and falling at night. The pressure readings were consistent with atmospheric trends, and the rain sensor reacted properly when it was raining.

The system's data were consistent with local climatic observations, validating its effectiveness in monitoring short-term environmental fluctuations. The capacity to monitor rainfall, for example, yielded valuable data for agricultural purposes, enabling farmers to forecast weather conditions influencing crop yields.

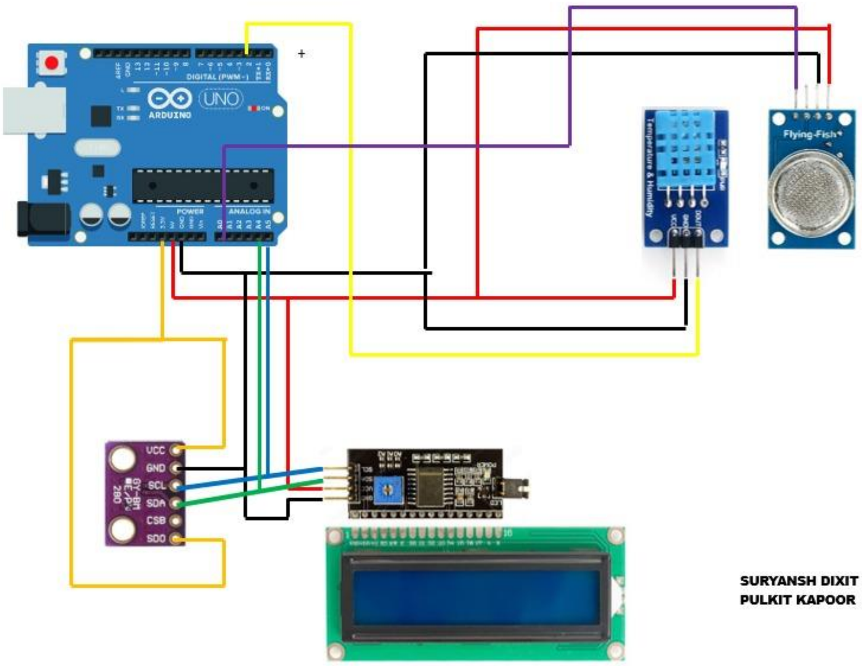
4.6. Future Challenges and Solutions

Despite its satisfactory performance, the system did have some problems. One of the problems was manual data extraction, which could be time-consuming for long-term observation. Future versions of the system can include cloud integration to save data automatically, which would be more convenient to analyze and visualize in the long term. Besides that, the provision of support for other sensors, such as a solar radiation sensor, would add more detail to the weather information. Increasing the power supply and the sensors' calibration process will further improve the efficiency and precision of the overall system.

CHAPTER 5: PROJECT



CHAPTER 6: CIRCUIT DIAGRAM



CHAPTER 7: ARDUINO CODE

sketch_weather_monitoring_system-Minor_Project.ino dht.h

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Adafruit_BMP280.h>
#include <DHT.h>

// --- Pin Definitions ---
#define DHTPIN 2       // Digital pin connected to the DHT sensor (D2)
#define DHTTYPE DHT11  // Type of DHT sensor
#define MQ135_PIN A0   // Analog pin connected to the MQ-135 sensor (A0)

// --- Sensor Objects ---
// Initialize the I2C LCD object
LiquidCrystal_I2C lcd(0x27, 16, 2);
Adafruit_BMP280 bmp; // I2C BMP280 sensor
DHT dht(DHTPIN, DHTTYPE); // DHT sensor

void setup() {
  Serial.begin(9600);
  Serial.println("Environmental Sensor Test");

  // Initialize BMP280
  if (!bmp.begin(0x77)) { // Check if BMP280 is found (common I2C address)
    Serial.println("Could not find a valid BMP280 sensor, check wiring!");
    // Consider adding LCD message here too
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0,0);
    lcd.print("BMP280 Error!");
    while (1); // Halt execution
  }
  Serial.println("BMP280 Initialized.");

```

sketch_weather_monitoring_system-Minor_Project.ino dht.h

```
  // Initialize DHT11
  dht.begin();
  Serial.println("DHT11 Initialized.");
  pinMode(MQ135_PIN, INPUT); // Set MQ135 pin as input
  Serial.println("MQ-135 Pin Initialized.");


  // Initialize LCD
  lcd.init();
  lcd.backlight();
  lcd.print("Initializing...");
  delay(1000); // Short delay to show message
}

void loop() {
  // --- Read Sensors ---
  float bmp_temp = bmp.readTemperature();
  float bmp_pressure = bmp.readPressure(); // In Pascals
  float dht_humidity = dht.readHumidity();
  float dht_temp = dht.readTemperature();
  int mq135_value = analogRead(MQ135_PIN); // Read raw analog value (0-1023)

  // Check if any DHT readings failed and skip display if so.
  if (isnan(dht_humidity)) { // You could also check isnan(dht_temp) if using it
    Serial.println("Failed to read from DHT sensor!");
    // Optionally display error on LCD
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("DHT Error!");
    delay(2000); // Wait before trying again
    return; // Skip the rest of the loop this time
  }
```

CHAPTER 8: CONCLUSIONS

Sensor Accuracy and Calibration

Sensor calibration is a critical part of ensuring the weather monitoring system's accuracy. All sensors, be they weather temperature, humidity, pressure, or rain sensors, need to be calibrated before deployment to ensure they are measuring the environmental parameters correctly.

The DHT11 temperature and humidity sensor, for instance, needs to be calibrated within a particular environmental condition in the expectation of reducing errors and enhancing reading accuracy.

The BMP280 sensor, used to detect atmospheric pressure, needs to be calibrated in order to adapt to varying environments. While the system can work in normal operations, it should be recalibrated periodically to achieve sensor stability and accuracy of data, especially when the system is operated in changing environmental conditions.

Data Storage: Cloud-based vs Manual Systems

For this project, the data is stored manually and not uploaded to cloud-based systems. While cloud storage offers the advantage of remote access to data and automatic backups, manual storage also has its benefits. In manual storage, the data that the weather monitoring system captures is stored locally, usually on an SD card or on a computer.

This gives more control over the data and prevents reliance on internet connectivity, which in certain situations proves to be transient in rural areas. Nevertheless, this approach has the drawback of involving manual data collection, which can prove to be time-consuming in the long run.

In future releases of the system, integration with the cloud might simplify storage and retrieval of data, the best of both worlds: quick access and local storage.

Future Development and Scaling

There are several chances to enhance and upgrade the weather monitoring system in the future. One of the most significant upgrades would be integrating cloud storage so that data is accessed instantly without any need to retrieve data manually.

This would provide users direct, remote access to weather conditions from anywhere in the world. Moreover, interfacing other types of sensors such as solar radiation sensors could provide users with yet more information concerning environmental conditions.

The system can also be scaled to monitor multiple locations at once, sending information from a group of sensors into a single database. Other enhancements can be to reduce power consumption to make the system more energy-efficient and enhance its user interface for easier data visualization and analysis.

By integrating these modifications, the system would be able to support a wider variety of applications, ranging from disaster preparedness to precision agriculture.

Observation

Since our Arduino Board only contains a single I2C port thus while using multiple I²C devices, make sure to not conflict their addresses.

The I²C address of BMP280 can usually be altered through the SDO/SA0 pin.

‘

REFERENCES

Book

2. S. K. Lally and J. A. Dyer, "**Smart Weather Monitoring System: IoT Based Real-Time Meteorological Data Collection",** Amazon.com, 2024
3. Arshdeep Bahga and Vijay Madisetti, "**Internet of Things: A Hands-On Approach"** , Orient Blackswan Private Limited - New Delhi, 2015

Online Documents

4. **IoT Based Weather Monitoring System Using Arduino**
   https://projecthub.arduino.cc/rajeshjiet/iot-based-weather-monitoring-system-using-arduino-a3334a
   A guide on how to build a fully functional weather station with sensors for temperature, humidity, and pressure using Arduino. It also explains data transmission and storage methods.

Libraries used

5. Library Used**: (Wire.h)** by Nicholas Zambetti**.**
6. Library Used**: (Liquid Crystal_I2C)** by Frank de Brabander
7. Library Used**: (Adafruit_BMP280.h)** by Kevin Townsend for Adafruit Industries
8. Library Used**: (DHT.h)** by Mark Ruys