DESIGN AND FABRICATION OF AN IoT-BASED RAIN AMOUNT MEASUREMENT

A Thesis
by

Shahriar Tanbir	Student Id: ME2102024483
Md. Azizul Haque	Student Id: ME2202027258
Md. Shamim Rana	Student Id: ME2101023075
Md. Masud Hossain	Student Id: BME2001020644

Department Of Mechanical Engineering
SONARGAON UNIVERSITY (SU)

May 2026

DESIGN AND FABRICATION OF AN IoT-BASED RAIN AMOUNT MEASUREMENT

A Thesis
By

Shahriar Tanbir	Student Id: ME2102024483
Md. Azizul Haque	Student Id: ME2202027258
Md. Shamim Rana	Student Id: ME2101023075
Md. Masud Hossain	Student Id: BME2001020644

Supervisor: Ibrahim Khalil Apurba
Lecturer

Submitted to the
DEPARTMENT OF MECHANICAL ENGINEERING
SONARGAON UNIVERSITY (SU)
In partial fulfillment of the requirements for the award of the degree
of
Bachelor of Science in Mechanical Engineering

May 2026

Acknowledgement

First, we would like to express our gratitude to Almighty Allah for giving us the strength and opportunity to complete our dissertation within the scheduled time successfully.

The presentation of the project paper represents the collaborative efforts and generous support of a number of individuals. The Department of Mechanical Engineering (ME), Sonargaon University (SU) provided important subject expertise and information technology services to carry out the project work successfully.

We feel proud to express our deep sense of gratitude to our reverend teacher, guide, and supervisor, for his day-to-day supervision, dexterous management, adept analysis, keen interest, optimistic counseling, and unremitting backup.

We are grateful to Ibrahim Khalil Apurba, the relevant lecturer in the Sonargaon University (SU) Department of ME, for letting us work on an intriguing notion and for resolving any issues that arose throughout its development. We express our gratitude to our administrator for his unwavering support, guidance, and invaluable advice, as well as for allowing us to collaborate on this project.

We are very much indebted to Prof. Md. Mostofa Hossain, Professor and Head, Department of Mechanical Engineering, Sonargaon University (SU), for including a dissertation as a course in B.Sc. in engineering to strengthen eligibility of students.

We would also like to thank those people of Department of Mechanical, Sonargaon University (SU), for their encouragement and help in our project work.

Abstract

Rainfall monitoring is essential for effective water resource management; however, traditional manual methods often result in inaccurate measurements, delayed responses, and poor recordkeeping. This project presents an IoT-based rain amount measuring and monitoring system that automatically detects rainfall, measures water level, stores data in the cloud, and displays real-time information through a mobile application. The system integrates a rain sensor and an HC-SR04 ultrasonic sensor, controlled by an ESP-8266 NodeMCU with Wi-Fi connectivity, to transmit data to a Firebase real-time database. An Android application developed using MIT App Inventor enables users to monitor live rainfall data and access historical records. A theoretical 9:1 area ratio between the rain capture surface and reservoir was applied, and calibration experiments improved accuracy by reducing measurement error to approximately 3%. Overall, the system provides a low-cost, reliable, and real-time solution for automated rainfall monitoring suitable for domestic and small-scale industrial applications.

Table of Contents

List of Tables

List of Figures

Notations

RTU-Remote Terminal Unit

GSM-Global System for Mobile Communications

IoT-Internet of Things

MTU-Master Terminal Unit

LAN- Local Area Network

IEDs -Intelligent Electronic Devices

PWM- Pulse With Modulation

SPI Pins-Serial Peripheral Interface

GPS-Global positioning system

Wi-fi-Wireless Fidelity

PCB-Printed circuit board

THD-Total Harmonic Distortion

PCS-Process Control Systems

TTL-Transistor Transistor Logic

SCK-Serial Clock

RTU-Remote Terminal Unit3

USB-Universal Service Bus

Chapter 1

Introduction

1.1 Introduction

In recent years, water has gained incalculable value. The diverse faces are the main reason for this circumstance. Water's various issues have been forerunners of population growth, aging infrastructure, obsolete technology, and groundwater contamination. These difficulties have pushed drinking water to the point where it is now considered a luxury rather than a need. This problem of scarce drinking water necessitates a feasible and effective strategy.

Recently, IoT has gained tremendous attention in the industry ranging from simple auto automation-based applications to sophisticated applications. The standard setting of water storage and circulation system in homes and industries consists of overhead storage, pipeline network, and underground storage tanks. IoT trend has opened up research areas, including investigating the systematic evaluation of water features by articulating its based sensor networks.

We designed a real-time wireless monitoring system for rainwater levels based on IoT and supported an alarming subsystem. The parametric water and wireless network were configured for accessible signal data collection and assessment. A real-time water level monitoring system was designed and implemented. An ultrasonic sensor was used to detect the desired parameter. If the water level reaches the pre-set parametric value, the signal will be fed in real-time to mobile. Depending on the sensor reading, the Arduino IDE program sends the data to ESP-8266 and through the nodeMCU board. Depending on the sensor reading, the Arduino IDE program sends the data to ESP-8266 and through the nodeMCU board. A practical and efficient approach is required to address the issue of overflowing rainwater. A Brief was considered to be addressing the issue of water allocation if it assessed the absolute or relative amounts of water used, or anticipated to be used, in any of these businesses.

1.2 Problem Identification

Project identification is a process in the initiating phase of the project life cycle for identifying a need, problem, or opportunity. Once identified, a project is initially documented, objectively defining what was identified. Every year, rainfall is a common occurrence in our country. But this system was controlled manually. This is the main problem in our lower region. This process is developed and monitored automatically by using IoT.

1.3 Objectives

The project's objectives are to

i. To determine when rain starts & stops.
ii. To determine the rain amount (how much rain falls in cm/sec).
iii. To develop an Arduino app to view and analyze all the data.

1.4 Necessity of Automation

i. In the textile business, automation refers to the tools and machines used to increase manufacturing efficiency. A higher-quality product, improved working conditions, and fewer hours of labor for the same amount of production are just a few benefits of automation.
ii. Automation, often known as automated control, is the process of operating machines and automobiles without human intervention by utilizing a range of control systems. Examples of this kind of machinery are industrial operations and boilers.
iii. Automation improves productivity and quality in an industrial context by reducing errors and waste, increasing security, and giving production processes greater flexibility. In the end, industrial automation improves safety, dependability, and profitability.

1.5 Advantages of Automation

i. Reduced operational costs: Robots can do jobs that would take three to five people to do. In addition to labor savings, automated tasks can save a significant amount of energy because they don't need to be heated as much. Robots improve part accuracy and speed up processes, which reduces waste in your company.

ii. Improved workplace safety: Automated cells have released people from harmful responsibilities. Your workers will appreciate that you protected them from the dangers of industrial labor.
iii. Faster ROI: Automation solutions that are tailored to your particular requirements and objectives soon pay for themselves through lower operational costs, shorter lead times, increased productivity, and other benefits.

1.6 Disadvantage of Automation

i. The topic of automation-induced job displacement has already been covered. Regardless of the potential social benefits of retraining displaced individuals for other employment, the person whose job has been replaced by a computer will almost always experience emotional hardship. In addition to losing their job, the person could have to relocate geographically. Having to relocate in order to find new job is another difficult circumstance.
ii. Automated equipment has some disadvantages, such as requiring a significant initial investment (the design, development, and installation of an automated system can cost millions of dollars), requiring more maintenance than a human-operated machine, and generally being less versatile.

1.7 Chapter Summary

This chapter focuses on the importance of IoT-based automation in overcoming the limitations of traditional, manually monitored water systems. It presents a real-time rainwater monitoring system using ultrasonic sensors, Arduino, and ESP-8266 NodeMCU to collect, transmit, and analyze rainfall data while providing alerts to prevent overflow. The chapter highlights how automation improves accuracy, efficiency, safety, and cost-effectiveness in water management, while also acknowledging challenges such as high initial costs and job displacement, ultimately emphasizing the role of smart, data-driven solutions in addressing water scarcity.

Chapter 2

Literature Review

2.1 Introduction

Water has become a luxury rather than a need in recent years owing to issues including population increase, deteriorating infrastructure, antiquated technologies, and groundwater contamination. Effective solutions are needed because of this scarcity, especially with the Internet of Things (IoT), which has led to a number of applications in water management. IoT technology was used to create a real-time wireless rainwater level monitoring system using an ultrasonic water level sensor and an alerting subsystem. An ESP-8266 and NodeMCU board receive data in real-time via the Arduino IDE, solving the problem of overflowing rainfall and supporting efficient water distribution.

2.2 Fundamentals of Automation

Any work may be readily automated. Microcontrollers have made it possible for tasks that were previously completed by people to be completed automatically. Every element of modern life has changed as a result of the entire substitution of human labor by machines, which has enabled the automation of our everyday tasks. They place a high value on convenience, safety, energy efficiency, and the ability to remotely monitor and control their home. The system should have intelligent automation features and intuitive control interfaces to seamlessly integrate their lifestyle.

2.3 Literature Review

The installation of a rain sensor intended for both daily usage and production applications is covered in this study. In order to allow users to gather and store rainwater for conservation, the project intends to detect rain and then sound an alert. This project promotes groundwater replenishment using underwater recharge technologies, demonstrating the sensor's usefulness in a number of industries, such as electronics, home automation, and irrigation. The system also includes a rainfall sensing circuit that is reasonably priced. [1]

As an early warning system against problems like dried clothing on a clothesline or possible floods, the Water Rain Sensor with Alarm is a gadget made to detect rainwater and notify users. It has sophisticated moisture sensors that continually check moisture levels in places like rooftops that are prone to water buildup. When the sensor detects rain, it sounds an alert and displays visual cues to persuade the user to take quick action. Additionally, the gadget has wireless connectivity for real-time notifications via cellphones, enabling users to remotely quiet the alarm and keep an eye on the amount of rain. [2]

Sun drying is an ancient technique used in homes, businesses, and farms to remove moisture from objects like clothing and some plants. Rainfall, however, can interfere with this procedure, thus a gadget that warns users of approaching rain is necessary to enable prompt material collection. This article describes the design and building of a rain detector that uses an Atmega 328p microprocessor. When it detects rain, the microcontroller uses a light-emitting diode to activate an alert and visual messages. The system's reactivity to various weather conditions, from no rain to severe downpour, is confirmed by testing findings, which show the strength of the rain on an LCD. [3]

The Rain alert Project is a useful project that detects rainfall and sounds an alert or buzzer in response. Given that water is necessary for life, this initiative highlights the need of water conservation. It acts as a catalyst to promote rainfall collection, which is essential for raising subterranean water levels via methods like underwater recharging. There are several uses for rainfall detectors, such as in autos, home automation, irrigation, and communication. For those who want to save rainwater for later use, this rainfall detector's circuit is easy to build, dependable, and reasonably priced. [4]

The rain sensor technology presented in this work aims to detect rainfall for useful applications in industry and daily life. The project highlights the value of conserving water and collecting rainwater for future use by using an A555 IC as a timer to sound an alarm. In order to enable acts like rainwater storage that can support groundwater recharge, the rain detector is made to notify users when it detects rain. The rainfall detection system is a cost-effective solution with a variety of uses, such as home automation and irrigation. [5]

Sun drying, a conventional technique for eliminating moisture from a variety of objects, is impacted by climate change, which has a substantial negative influence on quality of life. Rain frequently interferes with this procedure. In order to solve this, an Internet of Things (IoT) rain detecting gadget has been created. It uses a GSM notification system to notify users when rain is detected. The Arduino board uses a rain sensor module to monitor precipitation levels and detects environmental conditions. It then uses a GSM modem to transmit real-time text message notifications to a specified cellphone number. [6]

This study uses Internet of Things (IoT) technology, which is common in Smart Home applications, to create a rain detector that is coupled with an alarm system and the Blynk application. In order to let customers retrieve drying garments outside before becoming wet, the system detects rainfall and uses the Blynk app to deliver alerts and messages to users' mobile devices. The ESP32 Wifi Module and a rainfall sensor, which are coupled to enable system input, are the main parts of the project. The findings show two alternative ranges of values that successfully distinguish between "NOT RAINING" and "IT'S RAINING" circumstances. [7]

The use of IoT technology to detect rain and sound alerts is the main focus of rain alarm research. This development makes use of sensors and software to collect and distribute environmental and use data, enabling early problem identification in a variety of industries, including automation and healthcare. Using information provided to Firebase, the Bosch BMP280 environment monitor predicts rainfall. Users are informed when they require an umbrella by a magnetic switch sensor that senses temperature and air pressure. Rain sensors are essential for windshield wipers and automated irrigation. The study suggests design strategies for a rain detection system that is managed by an Arduino board that communicates with a rain control module and a rain sensor. [8]

The use of IoT technology to detect rain and sound alerts is the main focus of rain alarm research. This development makes use of sensors and software to collect and distribute environmental and use data, enabling early problem identification in a variety of industries, including automation and healthcare. Using information provided to Firebase, the Bosch BMP280 environment monitor predicts rainfall. Users are informed when they require an umbrella by a magnetic switch sensor that senses temperature and air pressure. Rain sensors are essential for windshield wipers and automated irrigation. The study suggests design strategies for a rain detection system that is managed by an Arduino board that communicates with a rain control module and a rain sensor. [9]

Even while technology is now a necessary element of everyday life, some people, like farmers, still avoid it. Reliable methods, particularly automated rain-protected drying buildings, are required to help these farmers. In order to automatically activate a protective cover over the crop during rains, this proposal suggests a system with a rain detecting mechanism that employs a sensor. To protect the crop from harm, a sophisticated microprocessor and a DC motor will interpret sensor data and operate the protective wrapper's deployment. [10]

The study presents a portable weather data collection station designed to enhance real-time weather monitoring, especially in rural areas. It addresses the challenges of traditional wired sensors, which require extensive installation and maintenance. By utilizing advanced wireless technologies, the station transmits data instantaneously or stores it online, employing DHT_11 and BMP_180 sensors for accurate measurements of temperature, humidity, and pressure. Data is sent via WiFi using NodeMCU V3 to Google Sheets, and MQTT protocol is incorporated for reliable communication. The system also integrates Google Maps for location tracking and MATLAB for thorough data analysis, providing a sustainable solution to meet the growing need for accessible weather information in remote regions.

2.4 Chapter Summary

We discuss many research articles on rainfall monitoring in this chapter. We attempt to provide a wide range of techniques that are either now in use or are being considered for use in rain fall measurement and monitoring. This study paper aids in the effective completion of our assignment. Studying research papers taught us about the benefits and drawbacks of a wide range of topics pertaining to our project.

Chapter 3

Introduction

3.1 Introduction

This chapter covers the hardware implementation, which is the main focus of our project. The project would not be feasible without appropriate implementation. Thus, we outline our hardware implementation and project success in this chapter.
Several devices with predetermined nominal values are required to complete the project. You won't receive the intended outcomes if the device ratings don't match. As a result, we have talked about the many gadgets we employed for the project and their specs.

3.2 Flow Chart of The Project

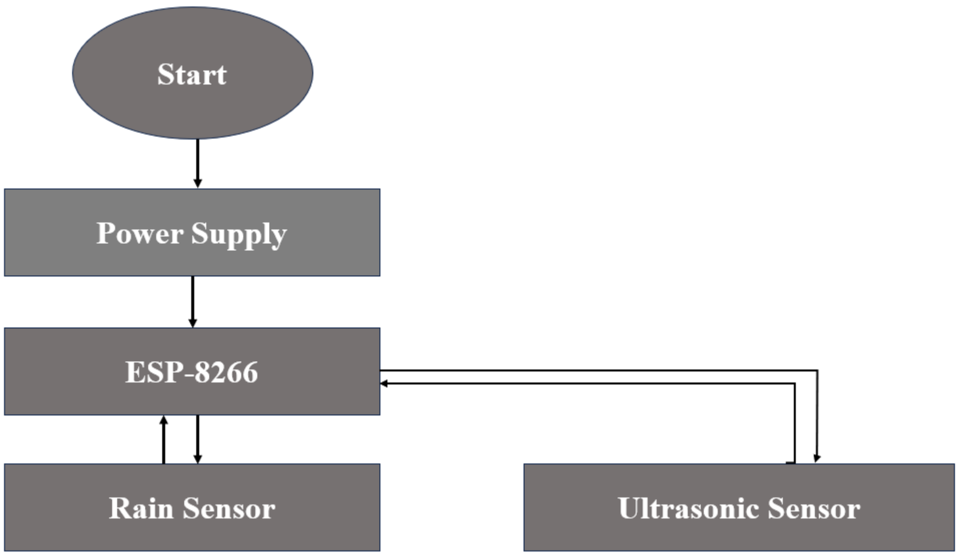


Fig 3.1 Flow Chart of the project

The flow chart shows that a Rain Sensor detects any rainy condition, and the Ultrasonic sensor measures the amount of rainfall. The ESP-8266 is used to send to cloud storage, receive, send, and keep an eye on the signal. All of the Rainfall amount data is to be saved in our database using an Internet of Things technology.

3.3 Block Diagram of the Project

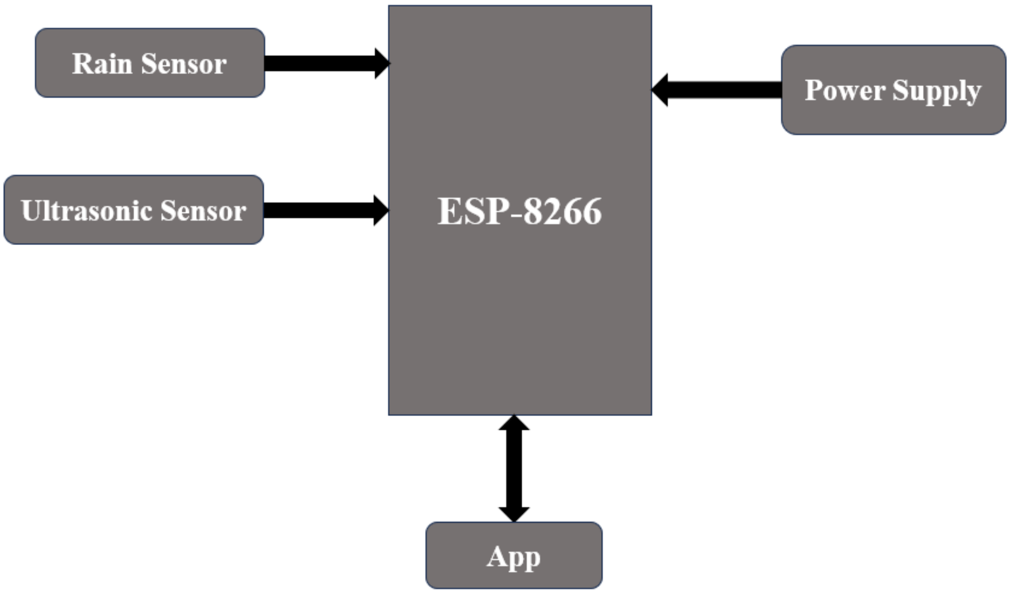


Fig 3.2 Block diagram of the project

The accompanying graphic shows the block diagram for our project. We can see several project components in this figure. These include the ultrasonic sensor, water sensor, water reservoir, power supply, and ESP-8266. The Ultrasinic sensor's value was recorded when the rain was starting. The rain sensor first sends a signal when the rain begins, in accordance with the ESP-8266's instructions. The microcontroller used the internet to connect to the Firebase database. As a result, the amount of rainfall is automatically measured, and the value is recorded. The user keeps an eye on his or her rainfall amount and may always observe the weekly rainfall record. Here is a brief overview of our project.

3.4 Circuit Diagram of our project

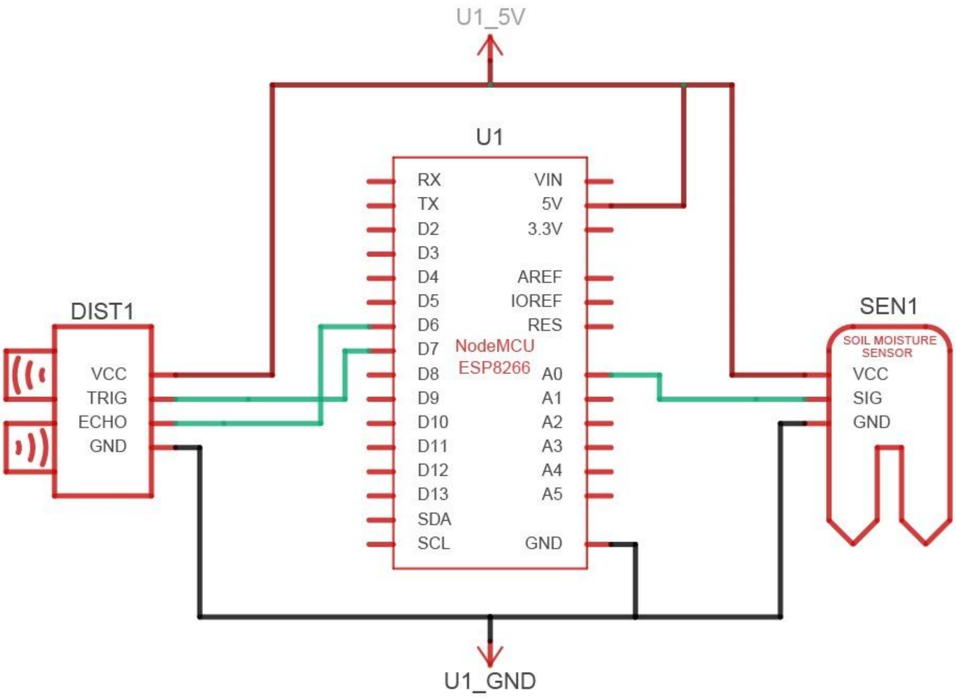


Fig 3.3 Circuit diagram of the project

A wire is connected between the relay module and the load in the circuit after the AC 220 V load is directly connected to the relay module's output side. 5 and 4.7 volts DC were produced from the remaining 220 volts AC. The ESP-8266 module is subjected to 5 volts DC. A current sensor was used to determine the light-taking power, and an ESP-8266 microcontroller was used to record the power. Additionally, the relay module runs AC 220 V. The relay receives signals from the Arduino and turns on when the load needs to be turned off automatically.

3.5 Design of Hardware

3.5.1 PCB Forming and Shaping

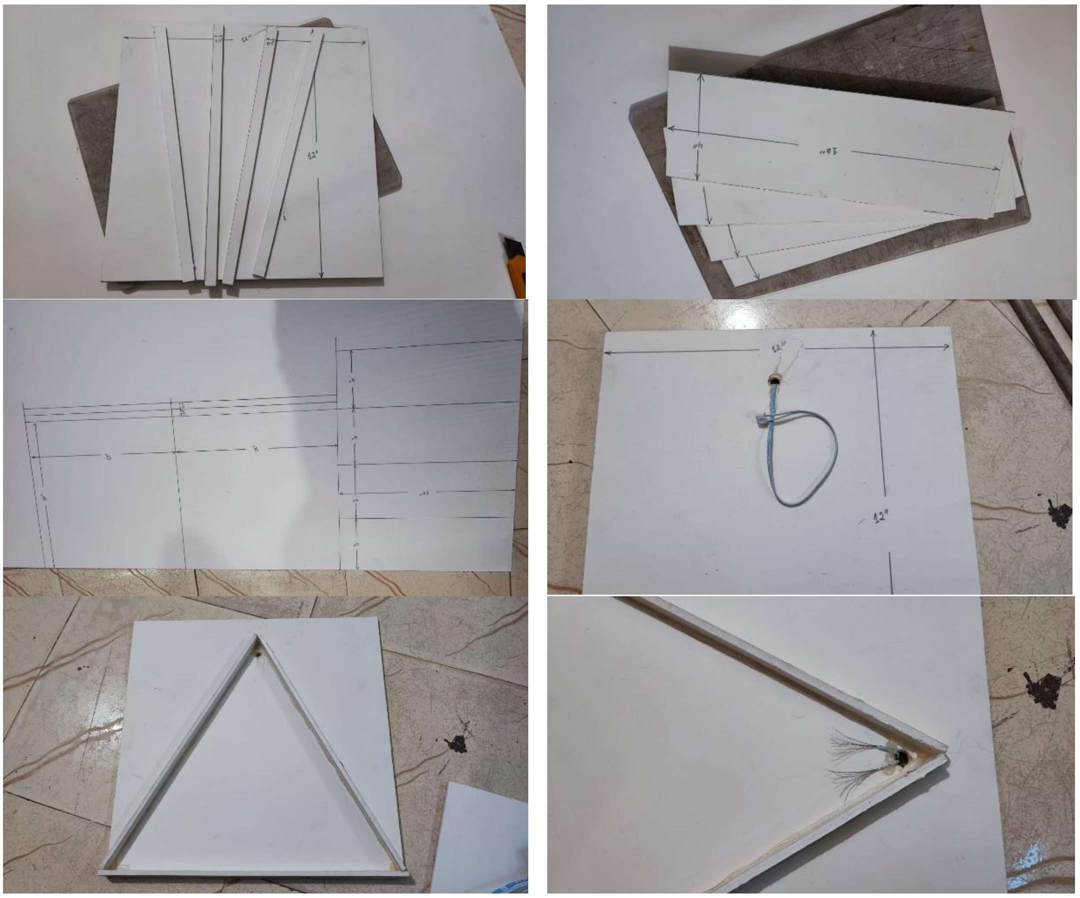


Fig 3.4 PCB Forming and Shaping

3.6 Hardware Setup

Fig 3.5 Final hardware setup

It is challenging to be fully informed on the project's hardware and software. The electronic parts have been positioned as optimally as feasible. The most crucial part of our project is the ESP-8266. We see this section as the core of the extension as it includes all of the computer program's information. This project also made use of an ultrasonic sensor and rain sensors. The ESP 8266 WiFi module is connected to the internet via the WiFi router.

Hardware elements:
The project used many hardware and software components, which are described below:

- ESP-8266
- Power Supply
- Ultrasonic Sensor
- Jumper Wire
- Rain Sensor
- PCB Board
- Water Reservoir
- Connecting Wire

3.7 Node MCU ESP 8266

NodeMCU could be a Espressif Frameworks Wi-Fi SOC (Framing on a Chip). The ESP8266-12E Wi-Fi module serves as its basis. It's a highly integrated semiconductor that provides a whole Web network in a small package. The Arduino IDE may be used to easily modify it over the USB harborage. With simple code, you may create a Wi-Fi connection similarly to how you would with an Arduino, define input and task legs as needed, and provide a web browser. It can be used to retrieve or send information, have a web garçon, or serve as a get-to-point or station. NodeMCU is an open-source, Lua-based firmware and development board specifically designed for Internet of Things (IoT) applications. It has firmware that manages grounding on the ESP-12 module and operates on the ESP8266 from Espressif Systems. It makes use of an ESP 8266mod Wi-Fi module. It is coupled to a microcontroller. It is often used to enhance the Web of Impacts (IoT). This WiFi module, which transmits all signals from the base station, looks a lot like the financial switch. Every piece of information should be stored in a standard database. For this reason, we used Firebase as the database. First and first, the WiFi module must process the supplied data or flag. For real-time observation, this data is stored on Firebase, which your phone will incontinently connect to.

Fig 3.6 ESP 8266 [12]

3.7.1 Power

The Arduino can be powered by a surface control restriction or by figuring out USB participation. The control source is properly selected. An AC-to-DC connection can also provide external (non-USB) control. You may use the DC Control Jack, the barrel jack, to control your Arduino board. Most of the time, a separator connector is connected to the barrel jack. Although the sponsor recommends maintaining the board between 7 and 12 volts, it may be powered between 5 to 20 volts. The controllers may warm up at 12 volts, and they may not function at 7 volts. The legs of the control are

i. VIN: by and large the input voltage leg of the Arduino board is utilized to supply input constraints from an exterior control source. The voltage has to be interior the degree indicated.

ii. 5V: This adherence of the Arduino board is utilized as a controlled control drive voltage and it's utilized to supply the board as well as onboard components.

iii. 3.3V: This adhere of the board is utilized to supply a drive of 3.3 V which is made from a voltage controller on the board

iv. GND: Inside the ESP- 8266 pinout, you'll find 5 GND legs, which are all associated. The GND legs are utilized to close the electrical circuit and deliver a common clarification reference position each through your circuit. Persistently make past any mistrustfulness that all GNDs (of the Arduino, peripherals, and variables) are related and have a common ground.

3.7.2 Analog IN

Only one analog pin of the ESP 8266 is used with the Analog Digital Converter (ADC). In addition to being used as analog inputs, these pins can also be used as digital inputs or outputs.

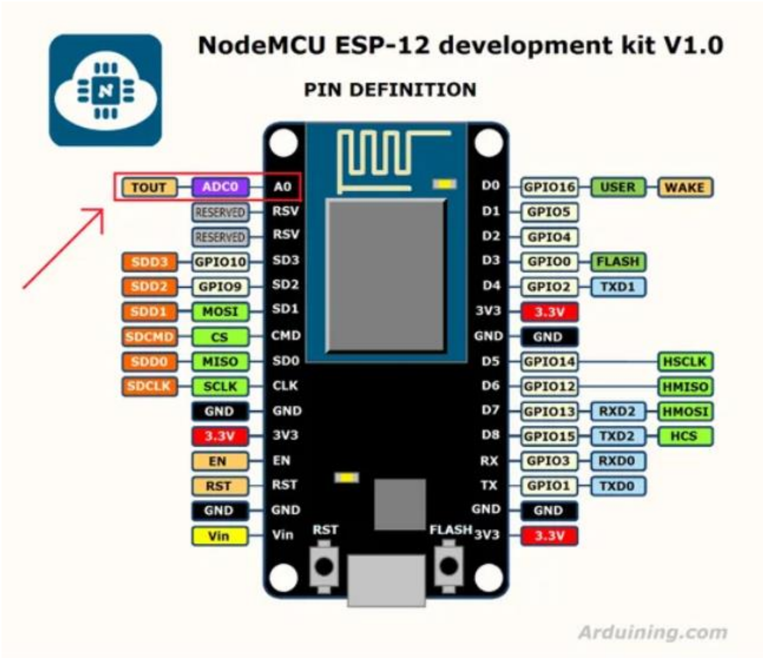


Fig 3.7 Pinout-Analog [13]

3.7.3 Pinout – Digital Pins

Legs 0- 13 of the ESP 8266 serve as advanced input/ issue legs.

Leg 13 of the ESP 8266 is associated with the erected-in Driven.

Within the ESP 8266- legs have PWM capability.

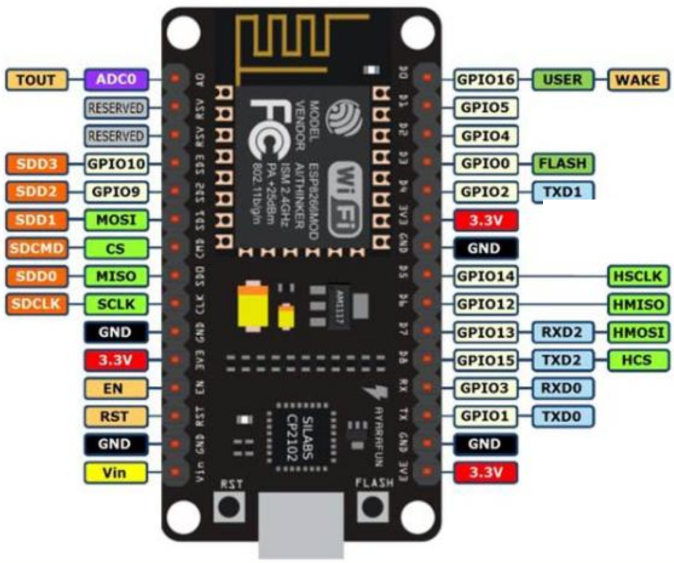


Fig 3.8 Pinout - Digital Pins[14]

It's critical to note that

- Each leg can allow/sink over to 40 mA most extreme. But the suggested current is 20 mA.
- The supreme most extreme current given (or sank) from all legs together is 200 mA.

3.7.4 Others Pins

i. **Pins:** These legs are too known as UART legs. It's utilized for communication between the Arduino board and a computer or other inclination. The transmitter Leg 1 and recipient Leg are utilized to transmit and concede the information resp.

ii. **Outside Hinder Pins:** This pin of the Arduino board is utilized to create the Outside barge and it's done by Legs 2 and 3.

iii. **PWM Legs:** The periodical legs of the board are utilized to change the computerized flag into an analog by changing the run of the palpitation. The Legs and 11 are utilized as a PWM leg.

iv. **SPI Legs:** This can be the periodical supplemental Interface leg, it's used to preserve SPI communication with the assistance of the SPI library. SPI legs incorporate.

v. **SS:** Leg 10 is utilized as a Slave Select.

vi. **MOSI**: Stick 11 is utilized as an Ace Out Slave In.

vii. **MISO:** Leg 12 is utilized as a Ace In Slave Out.

viii. **SCK:** Pin 13 is utilized as a periodical timepiece

ix. **LED Pin:** The board has an inbuilt Driven utilizing computerized leg- 13. The Driven shines as it were when the advanced leg gets to be tall.

x. **AREF Leg:** Typically an analog reference leg of the Arduino board. It's utilized to provide a reference voltage from an outside control drive.

3.7.5 Memory

Microcontroller	Family	Architecture	Flash	SRAM	EEPEROM
ESP 8266	AVR	Harvard	4 MB	64kb	1kb

Table 3.1 Memory of ESP-8266

3.7.6 Specifications

- Voltage: 3.3V.
- Soft-AP and Wi-Fi Direct (P2P).
- Current use ranges from 10 µA to 170 mA.
- Flash memory attachable: up to 16MB (512K is standard).
- Processor clocked at 80–160MHz.
- RAM: 32K + 80K.
- The number of GPIOs is 17 (multiplexed with other operations).
- One input from analog to digital with a resolution of 1024 steps.
- +19.5dBm is the 802.11b mode output power.
- Support for 802.11: b/g/n.
- Five TCP connections can be active at once.

3.8 Ultrasonic Sensor

The Ultrasonic HC-SR04 Sensor Module is a widely used sensor for measuring distance and detecting objects in a variety of applications. It is shown in Figure 3.9. It works in the same way that a radar system does. Ultrasonic sensors operate by producing high-frequency sound waves that humans can't hear. The transmitter emits a high-frequency sound pulse, which is received by the receiver when it reflects from any object's surface.

Fig 3.9 Ultrasonic Sonar Sensor [15]

The sensors detect objects in this manner. It can detect objects or measure distances between 2cm and 400cm. The ultrasonic sensor works in the same way that a bat's object-detecting system does. We can also claim that it works in the same way that a radar system does. The Transmitter (TX) and Receiver (RX) are the two primary components of the ultrasonic or HC-SR04 module (RX).

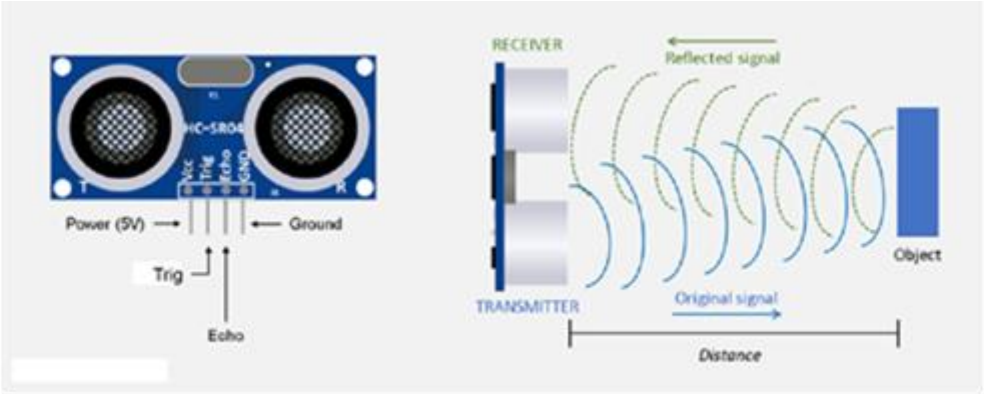


Fig 3.10 Distance measured by using ultrasonic sound[16]

3.8.1 Operating Condition

- Quiescent Current: <2mA
- Working Current: 15mA
- Effectual Angle: <15°
- Ranging Distance: 2cm – 400 cm/1″ – 13ft
- Resolution: 0.3 cm

3.8.2 Working System

Echo and Trig pin of the ultrasonic sensor at storage tank connected with D5 and D6 pin of NodeMCU. Again Echo and Trig pins of the ultrasonic sensor at the feed water tank are connected with the D7 and D8 pins of NodeMCU. The trig pin emits ultrasonic sound, which is received by the echo pin. To make the ultrasonic sound, we must first set the Trig pin to a High State for 10 seconds (microseconds). Where the distance between the sensor and the object is denoted by s. The sound speed in air is v = 0.034cm/s or 340 m/s. The time it takes for sound waves to bounce back from an object's surface is called t. Because time will be doubled when the waves travel and bounce back from the originating point, we must divide the distance value by two Power Supply − +5V DC.

3.9 Connecting Wire

JST Wire is used to establish a connection between the devices. ESP-8266 module, Rain sensor, and Ultrasonic sensor connection were done using this wire.

Fig 3.11 JST Connector [17]

3.10 PCB Printed Board

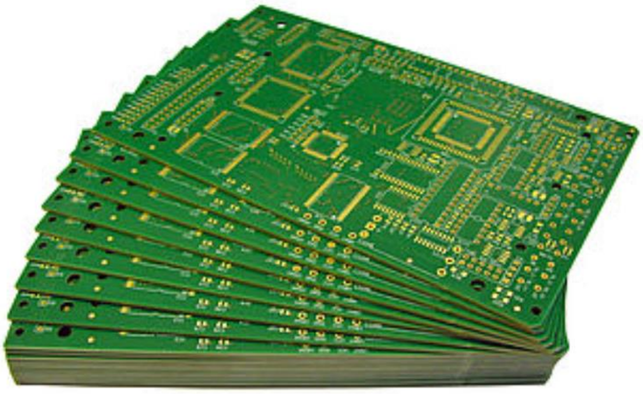


Fig 3.12 PCB Dot Varro Board [20]

Although "printed circuit board" is the most often used term, "printed wiring boards" or "printed wiring cards" are equally acceptable terms. Before the invention of PCBs, point-to-point wiring was a time-consuming method of building circuits. When wire insulation started to deteriorate and break, this resulted in fre quent failures at wire connections and short circuits. A simple printed circuit board consists of an insulating material that is flat and sturdy, with thin conducting components attached to one side. These electrical devices form squares, circles, and rectangles, among other geometric shapes. Different forms serve as component connection points, while long, thin rectangles serve as linkages.

3.11 Power Supply

One of the most useful power sources for our project is the 5V power. To convert a 50V AC or 240V AC input into a 5V DC output, a combination of transformers, diodes, and transistors can be utilized. We used a capacitor to clean up the DC voltage. With current technology, 5 volts is the ideal balance of speed, power consumption, and noise immunity. By using the same voltage to link circuits like sensors and other devices, the need for additional power supply may be reduced.

Fig 3.13 5V DC Adopter [18]

3.11.1 Working Principle

This is the main source of electricity for the project. All electronics need a 5-volt power supply to be activated. The GND site is connected to the power supply's positive side, while VCC, or 5 volts, is connected to its negative side. A USB connection is used to connect this portable power source. The power supply is suitable for our electric components due to its incredibly low current amp.

Chapter 4

Software and Apps Development

4.1 Software Development

Another crucial subject for this endeavor is software development. For this automated elevator monitoring system to work, the microcontroller has to be coded or programmed. The ESP-8266 is the microcontroller used in this project. The ESP-8266 controller was then controlled using the Arduino IDE. Using the App Controlled Lift mechanism, this step was flawlessly finished.

4.1.1 Software Arduino IDE

Let's take a closer look at the Arduino IDE's components after learning about them. You may do a number of actions, like creating and saving new files, by selecting the choices located near the top of the program screen. These button icons allow you to easily access some of the most frequently used functions. To check whether your code is error-free, click the verify button.

Fig 4.1 Arduino IDE [19]

You may transfer your code from your laptop to your Arduino and have it execute on your board by selecting Upload. You may input your program through the window, and you can read more about it in the message sections. As we work with the IDE, we'll go into more depth regarding messages, but for now, just know that they alert you to errors in your code and offer details like the RAM usage of the Arduino. Examine the buttons in the code editor's top row in more detail. You may quickly access the activities in the code window using these buttons. These actions involve making a new file, opening an existing file, and saving it. They also include uploading and testing your code on the Arduino board. Let's first discuss what a sketch is, and then we'll use each of these buttons. were kept without a file extension and produced using a text editor. The editor allows you to search for and replace text as well as cut and paste. When exporting and saving, the message section highlights errors and offers feedback. The Arduino Software (IDE) sends information to the terminal, including detailed error warnings.

4.1.2 Programming Interface

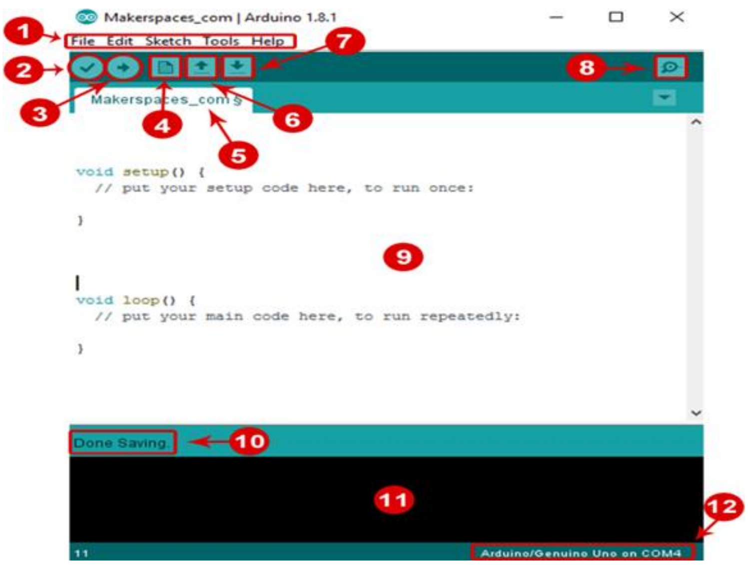


Fig 4.2 Sketch of Arduino IDE

Examine your surroundings and spend some time getting settled.

- Manu Bar: Provides you with the means to create and store Arduino sketches.
- Upload Button: Delivers the code to the attached board, in this example, the Arduino Uno. As the board uploads, the lights will flicker quickly.
- New Sketch: Brings up a new window with an empty drawing in it.
- Sketch Name: This is where the sketch's name appears once it has been saved.
- Open Existing Sketch: This lets you open a drawing that you've saved or one of the samples that are stored.
- Save Sketch: This preserves the drawing that is open at the moment.

- Serial Monitor: This will show the serial number of your Arduino when the board is attached.
- Coding Area: The sketch's code, which instructs the board on what to do, is composed in this section.
- Board and Serial Port: This indicates the board being utilized and the serial port to which it is attached.

You are now prepared to link your Arduino to your PC. Connect the USB cable's other end to the USB port on your computer and insert the first end into the ESP 8266. You must pick ESP 8266 under Tools, Board, then lastly once the board is linked.

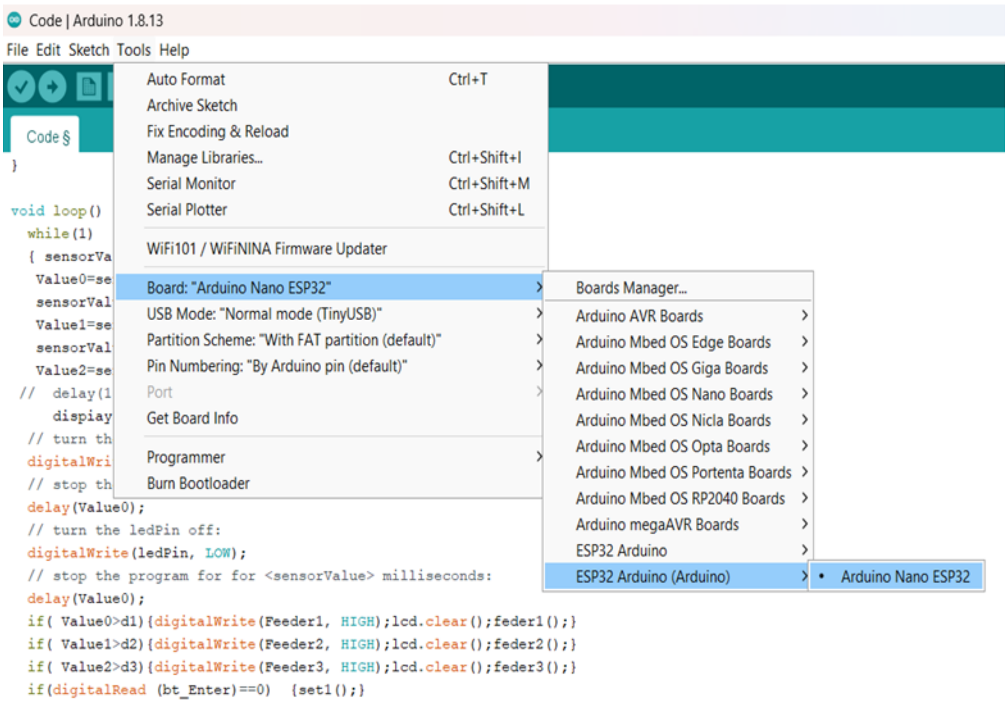


Fig 4.3 Arduino Board Selection

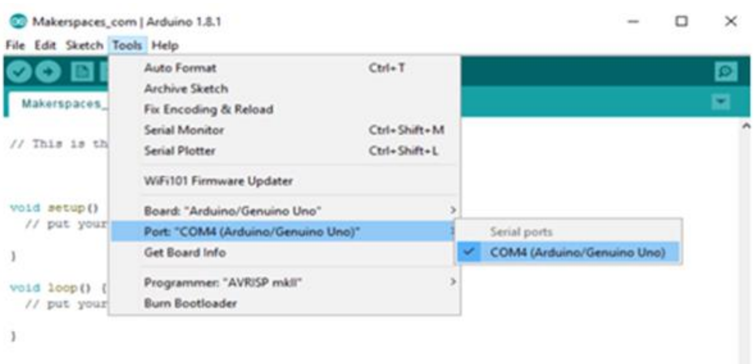
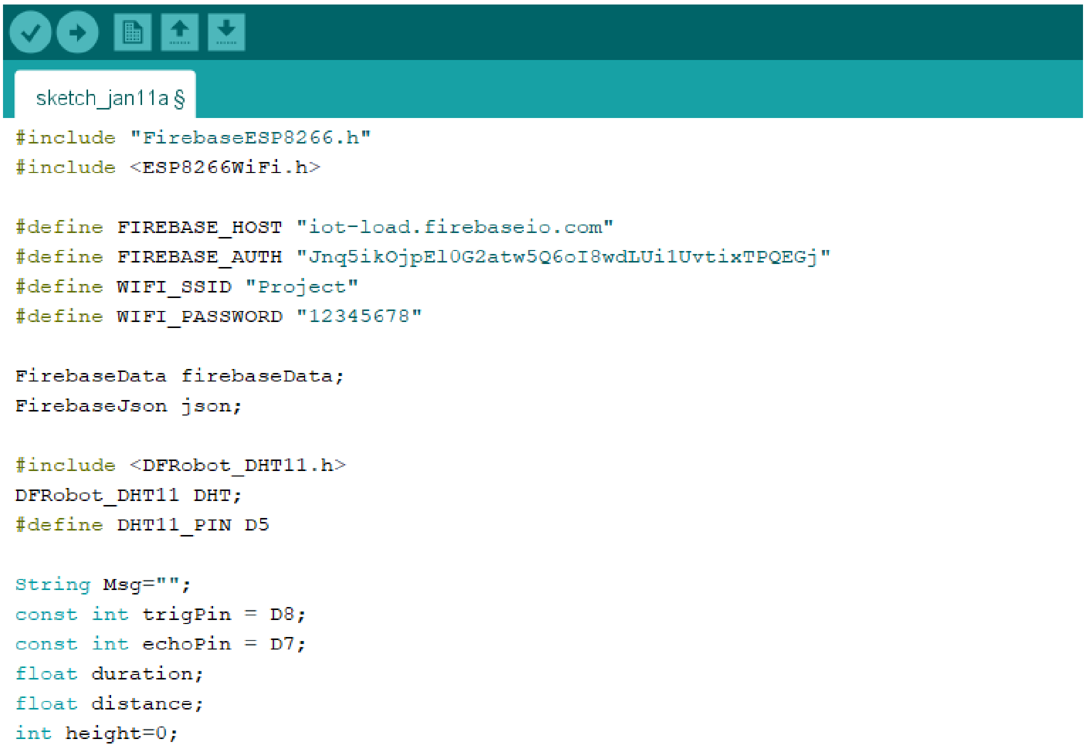


Fig 4.4 Arduino Port Selection

The Arduino needs to know which port on your PC you are currently utilizing. Go to Tools, then Port, and choose the port labeled Arduino to choose the port.

4.1.3 Arduino Variable Declaration and Library Include



```
#include "FirebaseESP8266.h"
#include <ESP8266WiFi.h>

#define FIREBASE_HOST "iot-load.firebaseio.com"
#define FIREBASE_AUTH "Jnq5ikOjpEl0G2atw5Q6oI8wdLUi1UvtixTPQEGj"
#define WIFI_SSID "Project"
#define WIFI_PASSWORD "12345678"

FirebaseData firebaseData;
FirebaseJson json;

#include <DFRobot_DHT11.h>
DFRobot_DHT11 DHT;
#define DHT11_PIN D5

String Msg="";
const int trigPin = D8;
const int echoPin = D7;
float duration;
float distance;
int height=0;
```

Fig 4.5 Arduino IDE screen Library Include and Void Setup

When coding, all of the libraries are first included in the Arduino IDE mane area. Different libraries are needed for different components since different components are used in different projects. We may choose any variable we wish to define as a variable after installing the required library. Here, we have added variables of the character, int, and string types.

4.1.4 Void Setup Function

File Edit Sketch Tools Help

sketch_jan11a §

```
void setup() {
  pinMode(D4,OUTPUT);
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input



  Serial.begin(9600);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");

  while (WiFi.status() != WL_CONNECTED)
  {
    digitalWrite(D4,1);
    Serial.print(".");
    delay(200);
    digitalWrite(D4,0);
    Serial.print(".");
    delay(200);

  }

  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  Firebase.reconnectWiFi(true);

}
```

Fig 4.6 Void Setup Function

4.1.5 Void Loop Function

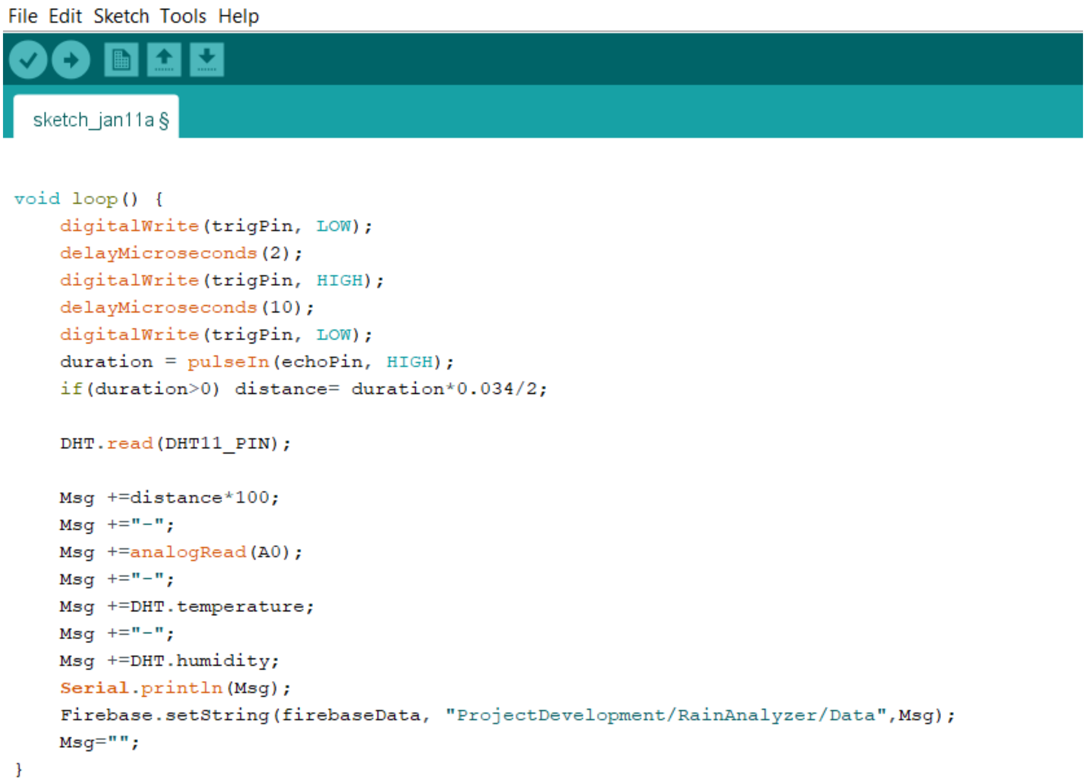


Fig 4.7 Void Setup Function

4.2 Internet of Things

The new paradigm known as the Internet of Things (IoT) has made living more technologically advanced than it was in the past. Smart homes, smart transportation, smart industries, smart libraries, smart pollution control, and strength saving are just a few of the advancements made possible by the Internet of Things. The use of IoT to decorate the time period has been the subject of several significant studies, research initiatives, and investigations. There are still many difficult situations and problems that need to be fixed before IoT may be used to its full potential. It is important to consider these difficult situations and problems from a range of IoT-related perspectives, including applications, difficult situations, enabling technology, social and environmental effects, etc. The item also highlights recent research and demonstrates how it adds to IoT-exclusive features. Additionally, the significance of vast amounts of data and how to assess them in the context of the Internet of Things have been discussed. This article may help readers and researchers better comprehend the Internet of Things and its potential applications in practical settings. IoT is gradually becoming an essential part of our lives that is evident everywhere. The Internet of Things (IoT) is a collective creation that combines large systems, smart frameworks, and intelligent gadgets and sensors. Additionally, it makes use of quantum and nanotechnology to reach previously unachievable levels of storage, sensing, and processing speed. With the growing involvement of IoT devices and the technology in daily ordinary lives, a remarkable transformation may be observed.Another essential IoT component is a Smart Health Sensing device (SHSS). The SHSS supports human fitness by means of gadgets and a small intelligent system.To measure and track certain fitness difficulties, health condition, the amount of energy spent in the fitness facility, etc., these gadgets may be utilized both indoors and outdoors.Additionally, it is becoming used to identifying urgent fitness circumstances in hospitals and trauma centers. Because of this, it has changed the condition of science in general by making it possible through complex gadgets and modern technology. Furthermore, scholars and IoT developers are always trying to make elderly and disabled people's current attire better. The majority of people are using such devices and systems because they are highly strong in terms of improvement fees and easily available within a daily charge budget.

4.3 Firebase

4.3.1 Using Firebase Functionalities in an Android App

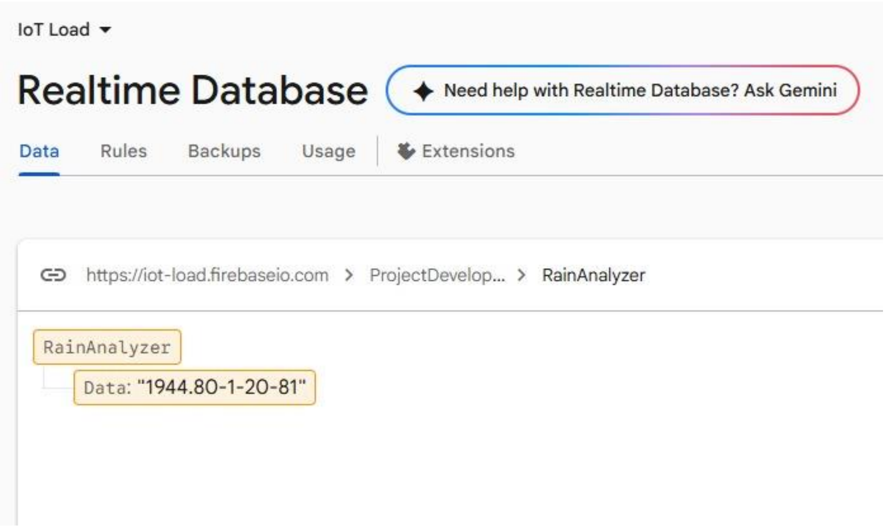


Fig 4.8 Real Time Database

All of Firebase's features may be used by the Android application with just a few lines of code. Topics including databases, storage, and authentication have all been discussed in this section. To learn more about the capabilities, click the link to the Google Firebase handbook in the reference section. This is how to use a few of the features.

4.3.2 Verification

The user can generate a login ID by using the following code once Firebase and authentication dependencies have been added to the Android application.

```
FirebaseAuthauth=FirebaseAuth.getInstance();
Auth.signInWithEmailAndPassword(email,password).addOnCompleteListener(new
OnCompleteListener())
{
@override
Public Void Oncomplete (task Task)
{ if(task.isSuccessful()) {Firebaseuser User=task.getResult().getResult().getUser();
String email=User.getemail();
//…
}}
}
```

4.4 Developing Apps

We developed an application to monitor this project.It is an Android application.Using this application is simple. This app was developed using the MIT App Inventor framework.Anyone can develop completely working programs for Android and iOS tablets, as well as iPhones and Android phones, using its intuitive graphical programming environment.Block-based coding programs foster people's intelligence and inventiveness.The four parking spots shown in this app are as follows. It is now in final beta testing as of July 8, 2019.It enables non-programmers to develop apps for Android and iOS, two operating systems (OS). It is free and open-source software, with the source code licensed under a Creative Commons Attribution ShareAlike 3.0 Unported license and the rest of the application under an Apache License 2.0. Users can now create Android apps by dragging and dropping visual objects into a graphical user interface (GUI) that resembles the programming languages Scratch and StarLogo.

Work is ongoing on an App-Inventor Companion, which would enable the app to run and debug on iOS devices.

Many different programming languages are used by App Inventor. A new programming language called "App Inventor Language" was developed by MIT App Inventor 2 blocks code. It has not yet been given a name.We're working on a textual version of this language called TAIL, or Textual App Inventor Language.

Many users will find this difficult to comprehend, yet the block code you see on your screen *is* your language's source code. When you save an. aia file (which is just a.zip file), the blocks are saved in an XML textual format that represents the same nested tree structure as the blocks on your screen. So, if you like to think of text as source code, this XML format is the App Inventor program's source code.

4.4.1 MIT App Inventor

A drag-and-drop interactive programming tool called MIT App Inventor may be used to create fully working Android mobile applications. App Inventor promotes a new type of private cell computing that allows people to design, create, and use—in my view—significant cell generation solutions for their daily lives in many circumstances. App Inventor promotes universal virtual literacy by allowing developers to concentrate on making informed programming decisions rather than coding language syntax thanks to its simple programming paradigm and progressive development features. Since it was moved from Google to MIT, certain improvements have been made, and research projects have started.

4.4.2 Drag and Drop Setup of the App

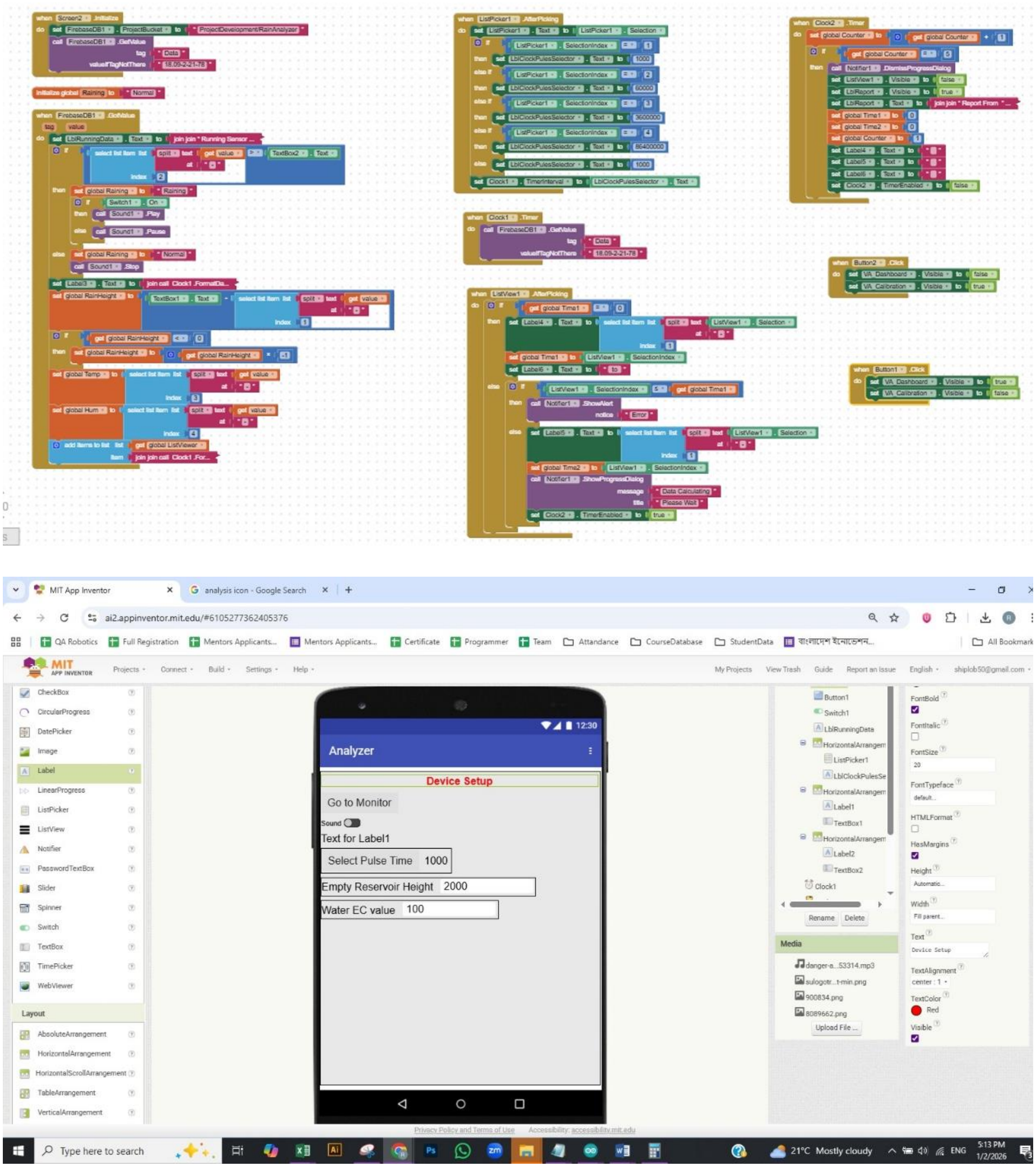


Fig 4.9 Drag and Drop code window

Before starting our major project, we created a few simple apps in App Inventor. The code and output part for the compass are seen above. The purpose of this block program is to inspect or create user-facing elevator buttons. This is linked by a microcontroller to the database.

Chapter 5

Result And Discussion

5.1 Data Calculation

Water Capture Area, Ac = (√3/4) a^2	Water Reservoir Area, A = b^2
Here, a = 12inch = 30.48cm	Here, b = 4inch = 10.16cm
Ac = 929cm^2	Ar = 103cm^2

Ratio of Ac:Ar = 9:1

If there is 1 millimetre of rain, our machine will display 9 millimetres.

We need to calibrate it.

We used a watering can in the garden for experimental purposes.

Our reservoir capacity is about 3 litres.

So, we experimented with increasing and decreasing the time by adding 2.5 litres of water to the gardening jar.

Exp. No.	Duration in (sec)	Water in (ml)	Height from App	Exp. Result in (cm)	Theoretical Result	% of Error
1	318	2500	26.0	2.8889	2.696871629	7.12
2	296	2500	27.5	3.0556	2.696871629	13.30
3	188	2500	25.0	2.7778	2.696871629	3.10
4	276	2500	28.0	3.1111	2.696871629	15.63
5	241	2500	26.8	2.9778	2.696871629	10.41
6	170	2500	27.2	3.0222	2.696871629	12.06
7	160	2500	25.5	2.8333	2.696871629	5.06
8	102	2500	28.5	3.1667	2.696871629	17.42
9	82	2500	26.3	2.9222	2.696871629	8.35
10	55	2500	27.0	3.000	2.696871629	11.23

Table 5.1: Collecting Data

Here,

Duration in sec: We have been pouring water for as long as we can.

Water in ml: The amount of water supplied.

Height from App: The information displayed in the app from the machine database

Exp. Result in cm: The data obtained from the app had to be divided by 9 to be presented realistically.

Theoretical Result: Theoretically, the results should be displayed.
% of Error: The difference between theoretical and practical experiments is the percentage change.

5.2 Result Analysis

The system was theoretically designed to use a 9:1 area ratio, meaning that 1 mm of actual rainfall should produce 9 mm of displayed height. However, experimental results showed noticeable variations due to practical factors such as uneven water pouring, minor losses, and sensor limitations. The percentage error ranged from 5.06% to 16.60%, indicating that theoretical assumptions alone are insufficient. After calibration, the effective ratio was found to be approximately 10:1, which significantly reduced the average error to around 3%. This confirms that experimental calibration is essential for improving accuracy in IoT-based rainwater measurement systems and makes the proposed system reliable for practical applications. In paper [11], we see that there are many analytic calculations they made, but in the real experiment we found error not the exact accurate value.

Chapter 6

Conclusion And Future Scope

6.1 Conclusion

- The smart rain measure is a cutting-edge technology. This technology's key benefit is that it can accurately calculate the height of rainfall in the rainy season and the reservoir tank.
- The adoption of a smart water level indicator system helps to develop the fourth industrial revolution. Compared to the manual system, this system offers the advantage of monitoring via a mobile The adoption of a rain measurement system helps to develop the fourth industrial revolution. Compared to the manual system, this system offers the advantage of monitoring via a mobile app.
- The owner of the house should always be alert when raining to an area. Now this data will be saved on the mobile via IoT by using the nodeMCU, ultrasonic sensor, database, and mobile app.
- The height of water was shown in the Apps display so that the operator can determine the condition of the rainfall level in the area.
- Finally, the project has implemented of rain measuring and monitoring system. This project can be developed with the help of our supervisor.

6.2 Recommendations And Further Study

- We can also include the GSM-based system, where the message will be sent to the particular authorized person when the water level is higher than the required level.
- Arduino GPS shield can be integrated into the system to obtain the location data of the water source dynamically.
- The PH sensor will also be attached in the future. So that the properties of water will also be known.
- Website base data communication system will also develop for this project. So the result show in android apps at a time in website.

REFERENCES

1. Dhumal, K., Meshram, T., Padol, A., & Yeole, G. (2022). Electrical Rain Detector Using Arduino Uno with LCD and Buzzer. *International Journal of Advanced Research in Science, Communication and Technology*, 444–447. https://doi.org/10.48175/ijarsct-2966

2. Ahmad, N. A. (2023). Development a Rain Sensor With Alarm System. Multidisciplinary Applied Research and Innovation, 4(4), 199-206.

3. Iyen, Cookey, Benedict Ayomanor, Abubakar Orume, Samuel Saleh, Simon Jaafaru, and Bunmi Jacob Akeredolu. "Design and construction of a rain detector with an alarm system." FUW Trends in Science Technology Journal 5, no. 3 (2020): 686-690.

4. Baghel, Rajendra. "Automatic Rain Sensor Alarm." Academia. Edu (2019).

5. Hoor, Alaa. "Rain Sensor Alarm Project." (2021).

6. Kennedy, GRACE CHINYERE, and T. Tammy. "Design of an Internet of Things Rain Detector Device with GSM Notification." *Iconic Research and Engineering Journals* 6, no. 3 (2022).

7. Zulkiflee, Ahmad Luqman Bin, Puteri Aisyah Nur Dinie Binti Sahadan, Nur Fathiah Binti Rosmadi, and Muhammad Faizal Bin Ismail. "Development of Rain Detector System using ESP32 With Alarm and Blynk Application." Multidisciplinary Applied Research and Innovation 3, no. 1 (2022): 482-489.

8. Abdullahi, Abdulrahman Yusuf, Mukhtar Ibrahim Bello, Sadiku Aminu Sani, and Ahmad Baballe. "Components utilized in the rain detector device's implementation." Journal homepage: https://gjrpublication. com/gjrecs 3, no. 04 (2023).

9. Devi, P., U. A. Jokindher, S. KARTHIK, and R. KEERTHIVASAN. "Automatic Rain Sensing Wipers using Arduino." *International Research Journal of Engineering and Technology* 9, no. 1 (2022): 38-41.

10. Goud, P. Goutham, N. Suresh, E. Surendhar, G. Goutham, and V. Madhu Kiran. "Rain sensor automatically controlled drying shed for crop yield

farmers." International Research Journal of Engineering and Technology (IRJET) (2010).

11. Sampath, W. M. A., R. L. W. Koggalage, K. Y. R. Amaradewa, and H. G. A. Sampath. "IOT Based Real-Time Portable Weather Data Collecting Station." In RESEARCH SYMPOSIUM, p. 185.

12. https://share.google/N7eTmWxxXTWDXVWon

13. https://share.google/5ppK7xcWGQtSJ9BJC

14. https://share.google/M3kjWSCr2dWF8htSH

15. https://share.google/mN7dIDT5t5ngGGhwe

16. https://share.google/AVsDBuSVSNLJZoxPs

17. https://share.google/6liq6f00yO9FN1JYF

18. https://share.google/q6IsbXj9DRjzQ6LEl

19. https://share.google/kfj7IQYEOVakF3X2P

20. https://share.google/O8IYFsQGDYvU2nWxV

APPENDIX

A. Programming Codes for Proposed Project

```
#include "FirebaseESP8266.h"
#include <ESP8266WiFi.h>

#define FIREBASE_HOST "iot-load.firebaseio.com"
#define FIREBASE_AUTH "Jnq5ikOjpEl0G2atw5Q6oI8wdLUi1UvtixTPQEGj"
#define WIFI_SSID "Project"
#define WIFI_PASSWORD "12345678"

FirebaseData firebaseData;
FirebaseJson json;

#include <DFRobot_DHT11.h>
DFRobot_DHT11 DHT;
#define DHT11_PIN D5

String Msg="";
const int trigPin = D8;
const int echoPin = D7;
float duration;
float distance;
int height=0;

void setup() {
  pinMode(D4,OUTPUT);
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
```

```
  Serial.begin(9600);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");

  while (WiFi.status() != WL_CONNECTED)
  {
    digitalWrite(D4,1);
    Serial.print(".");
    delay(200);
    digitalWrite(D4,0);
    Serial.print(".");
    delay(200);

  }

  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  Firebase.reconnectWiFi(true);

}


void loop() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
```

```
    if(duration>0) distance= duration*0.034/2;

    DHT.read(DHT11_PIN);

    Msg +=distance*100;
    Msg +="-";
    Msg +=analogRead(A0);
    Msg +="-";
    Msg +=DHT.temperature;
    Msg +="-";
    Msg +=DHT.humidity;
    Serial.println(Msg);
    Firebase.setString(firebaseData, "ProjectDevelopment/RainAnalyzer/Data",Msg);
    Msg="";
}
```