# DATABASE MANAGEMENT SYSTEM LAB
# MINI PROJECT
## Course Code: 310246
**(2019 Course)**

## Third Year Engineering

Year 2021 - 2022

Group ID: 03


Team Members: 1. MAYUR MHATRE (TACO19130)

2. DIVYA KHESE (TACO19139)

3. GAURAV GHYAR (TACO19141)


Project Title : Raspberry Pi 4 And NodeMCU Esp32 Based Air Quality Monitoring System with Cloud Integration

Name of Mentor: Dr. MANISHA BHENDE

**A PRELIMINARY REPORT ON**

# RASPBERRY PI 4 AND NODEMCU ESP32 BASED AIR QUALITY MONITORING SYSTEM WITH CLOUD INTEGRATION

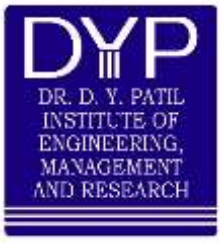
SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE ACADEMIC

OF

## SECOND YEAR OF COMPUTER ENGINEERING

**SUBMITTED BY**


| | |
|---|---|
| **MAYUR MHATRE** | **ROLL NO. TACO19130** |
| **DIVYA KHESE** | **ROLL NO. TACO19139** |
| **GAURAV GHYAR** | **ROLL NO. TACO19141** |



## DEPARTMENT OF COMPUTER ENGINEERING

## DR. D.Y.PATIL INSTITUTE OF ENGINEERING, MANAGEMENT & RESEARCH

**AKURDI, PUNE 411044**

# Table of Contents

# 1. TITLE OF THE PROJECT

## RASPBERRY PI 4 AND NODEMCU ESP32 BASED AIR QUALITY MONITORING SYSTEM WITH CLOUD INTEGRATION

# 2. ABSTRACT

A kind of wireless remote monitoring and control system based on Raspberry Pi 4 and NodeMCU ESP32 is presented in this paper. This design has the advantages of stability and ultra-low power consumption. By transplanting MQTT on the ESP32, to achieve a homemade weather station Internet connection and automatically obtain an IP address, the user through the mobile phone or a PC web browser to access the system WEB server through the web page display the temperature, humidity, Gas Concentration, PM2.5 and other sensor data, and network gateway can control sub-node peripherals use the control instruction which be uploaded to be sent to the sub-node, achieving wireless LAN to the Internet for remote monitoring and control system by MQTT protocol, the webpage will display the dashboard consisting Real-Time Monitoring graphs. Meanwhile, the system can also be used as a client, the collected data is uploaded to a common MySQL database.

**Keywords:**
Raspberry Pi, Arduino, NodeMCU, ESP32, MQTT, AQMS, Air quality monitoring system, MQ2 sensor, MQ9 sensor, MQ135 sensor, GP2Y PM2.5 Sensor

# 3. INTRODUCTION

## 3.1 INTRODUCTION

In today's world, air pollution, climate change, and its consequences are of a great concern to the environmentalists and climate change scientists. Emission of various poisonous gases from industries and vehicles are not only hazardous for the terrestrial organism, but the marine life is also getting adversely affected. As the world's population is becoming increasingly urban, the cities are under pressure to remain livable. Health problems arising due to poor air quality are in increase like stroke, heart diseases, lung cancer, respiratory diseases including asthma. Poor air quality poses a significant risk to the vulnerable section of the society such as children, asthmatic, pregnant women, and the elderly persons. As per WHO statics, millions of premature death cases are reported due to air pollution every year worldwide. Due to this, in recent years, the air quality of the cities has become one of the major cause of concern around the world. Thus, it is necessary to constantly monitor the air quality index of a city to make it smart and livable.

Around the world, governments are building the smart cities to keep a check on these problems and provide a healthy life for its inhabitants. The Indian government is in the process to build 100 smart cities by 2050. These cities will utilize advanced communication network, WSNs, and intelligent system to solve future challenges and create new services. Although the Indian cities have installed real-time air quality monitoring systems in Delhi and other cities, low-cost IoT enabled WSN technology is the future for the coming smart cities around the world.

Real-time monitoring of the air quality requires the live data transfer between the devices over the internet and it can be visualized using an Android Application. It reduces the mobilization of system hardware at different locations, which is cost efficient as only one-time installation cost is involved. In the Internet of Things (IoT) based applications, Raspberry pi 3B, having a wide range of specifications, is the mainstay of the system. It not only gathers data from various sensors via an inbuilt Wi-Fi module but is also responsible to send the recorded data to the ThingSpeak, an open source cloud platform on which data can be stored and retrieved via hypertext transfer protocol (HTTP) over the internet. ThingSpeak acts as a platform to store real-time sensor data and also used to plot graphs, charts, create plugins and apps for collaborating with web services, social network and other application program interface (API). Once signed in, a channel is created with a unique Channel ID. The primary feature of ThingSpeak is the term Channel in which there are eight fields for data storage, three fields to store latitude, longitude and elevation and one field to write a short message to describe the data. Once the channels are created in ThingSpeak, the data can be implemented and alternately one can process and visualize the information through various resources and platform, one of them being on an Android Application, designed in Android Studio.

The increased demand for service over the internet has necessitated the data collection and exchange in an efficient manner. IOT has promised the ability to provide the efficient data storage and exchange by connecting the physical devices and vehicles via electronic sensors and internet.

# 4. SOFTWARE REQUIREMENT SPECIFICATION

## 4.1 Introduction:

### 4.1.1 Purpose:

The purpose of the project - Raspberry Pi 4 And NodeMCU Esp32 Based Air Quality Monitoring System with Cloud Integration is to create awareness about one's surrounding air quality. Since, Air Quality Index is rising each and every year, the air we breathe is getting polluted. Risk of symptoms and diseases increase. Asthma, difficulty in breathing, lung cancer, ischemic heart disease, stroke, chronic obstructive pulmonary disease (COPD) and acute lower respiratory infections in children are the most common disease caused by air pollution.

### 4.1.2 Document Convention:

The document is organized as follows:

- Introduction
- Product Description
- System Features
- Hardware Features
- Software Diagram

### 4.1.3 Intended Audience & Reading Suggestions:

The intended audience is:

- Team Members
- Mentors
- Customers

### 4.1.4 Project Scope:

This document provides a general information about the project and addresses all requirements defined and discussed with the team members. The functional requirements will be described by means of hardware features and software diagrams. The document provides information about project and its diagrams.

### 4.1.5 References:

Air Quality Monitoring System, Raspberry Pi, NodeMCU ESP32,
MQ2, MQ9, MQ135, GP2Y Dust Sensor, DHT11 Sensor

## 4.2 Overall Description:

### 4.2.1 Product Perspective:

The intended stakeholders of the application are on one side the common citizens which are interested in knowing about the air quality of their home town and keep track of harmful levels of pollutants. On the other side NGO organizations, interested in environmental sustainability, would use the application as a practical tool to inspect air quality in near-real time over the Earth.

### 4.2.2 Product Features:

The Product includes five sensors to collect real-time data and sent it to the cloud. Over the Internet, the user can view various info about their surrounding using the dashboard. The dashboard views Graphs and percentage of gases in the region. Using such data, the user can be alerted, if an emergency like a Gas Leak occurs.

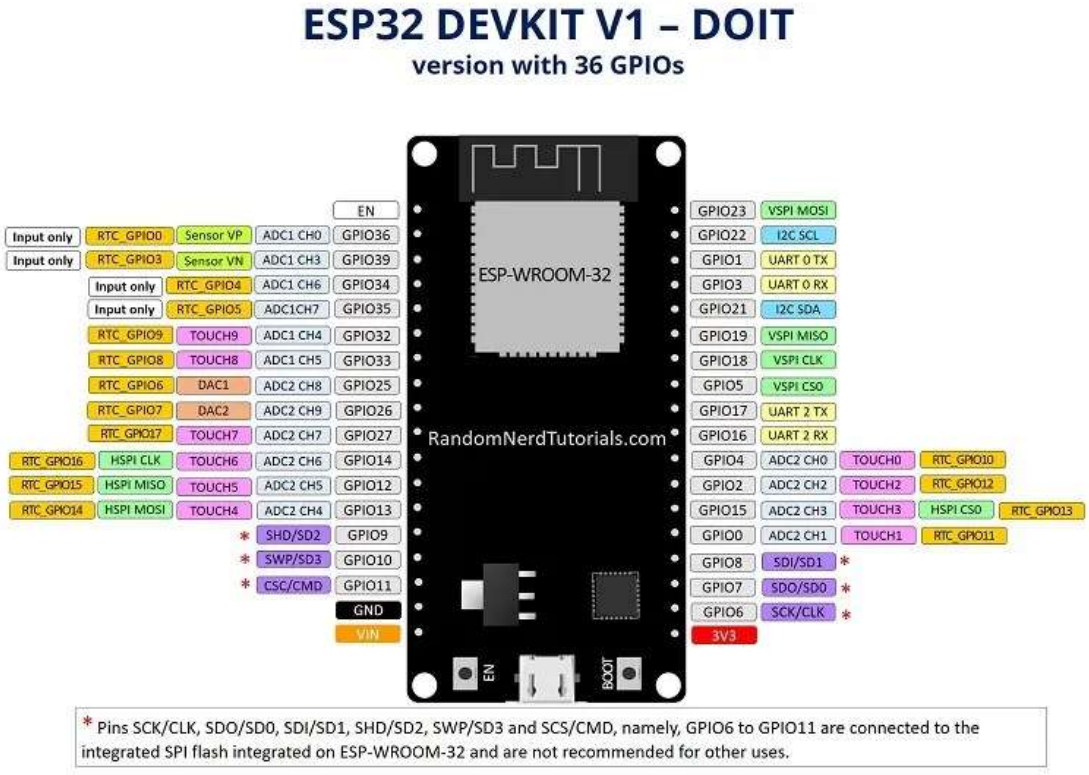
### 4.2.3 User Classes and Characteristics:

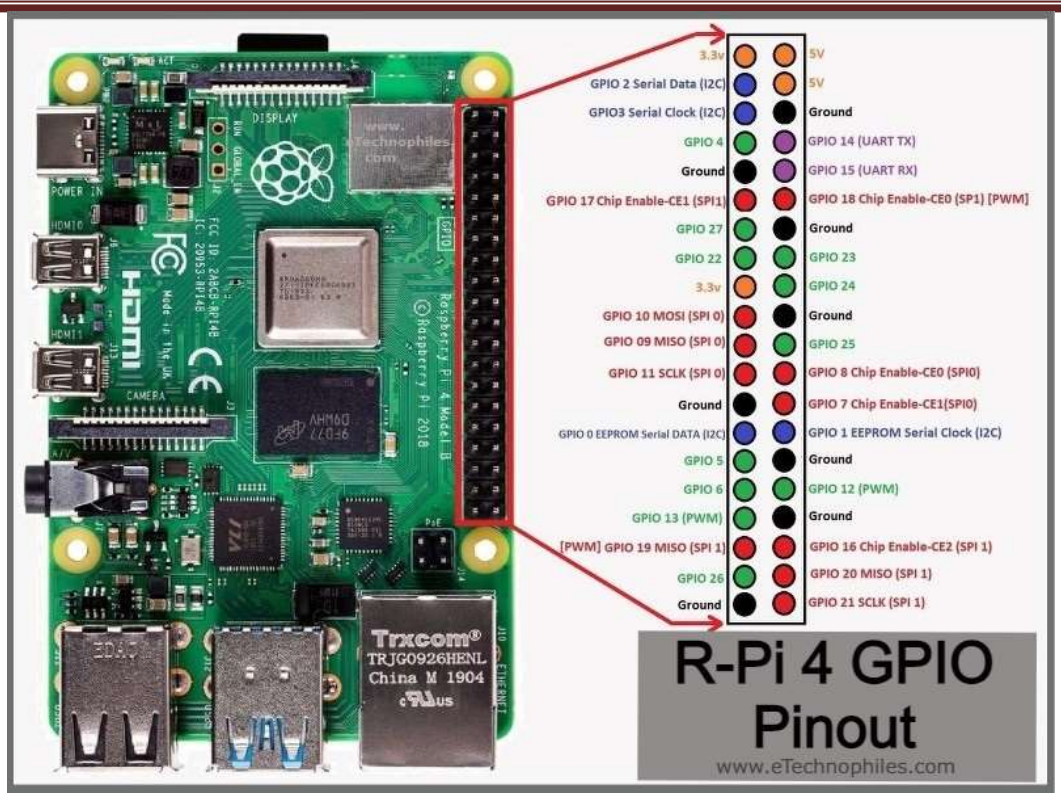
- Sales People
- IT department

### 4.2.4 Operating Environment:

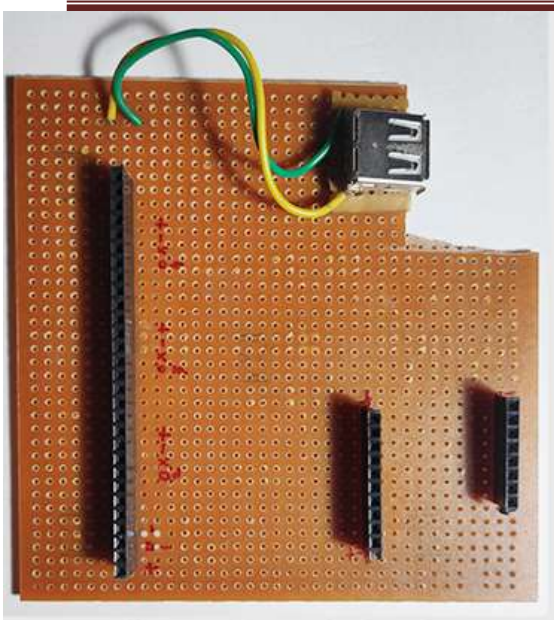
Home Environment, Room, Office, Work Area

### 4.2.5 Design and Implementation Constraints:

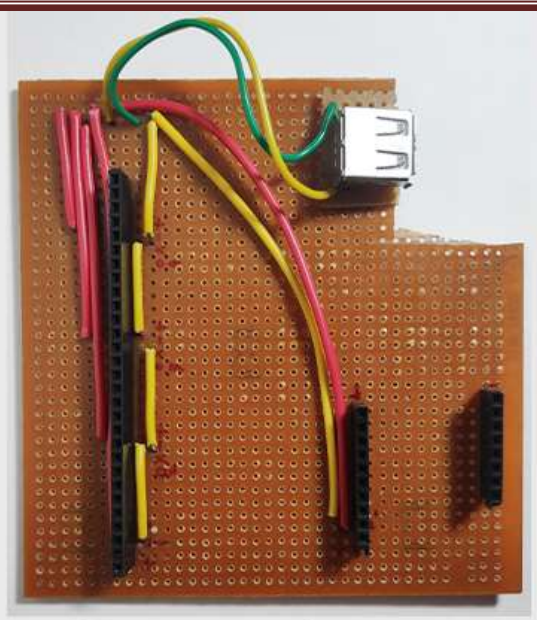




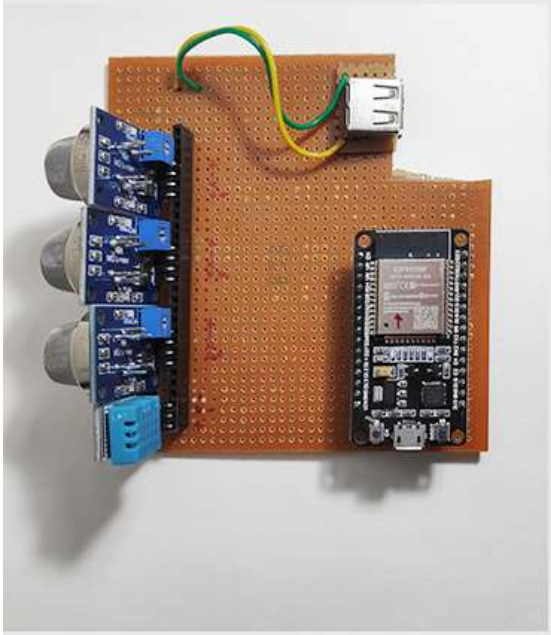

NodeMCU ESP32 and Raspberry Pi 4 Pinout diagram that explains about GPIO output of both controllers.



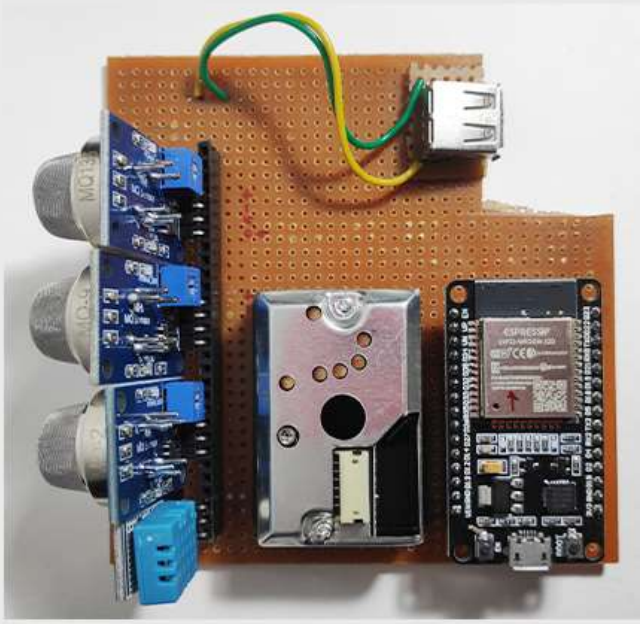
Installation of Female headers



Soldering Wires and Connections



Testing Sensors and ESP32



AQMS Prototype with PM2.5 Sensor
and
Air Quality Sensor Array

### 4.2.6 User Documentation:

User feedback from using the dashboard for a day was overwhelming with the amount of accuracy the sensor was providing. Cheap and expansive, the sensor array can detect many types of gases within a given region.

### 4.2.7 Assumptions and Dependencies:

Assumptions: The user has to keep the Master server – ON all the time as it records the data sent by nodes. Nodes can stay at a range of 50M from the Master server for full efficiency. User needs to have good Internet Router for maximum range and data transfer speed.

Dependencies: 300mbps Router (Internet access for cloud networking), two 5v 2A adapters for powering the Master server and nodes.

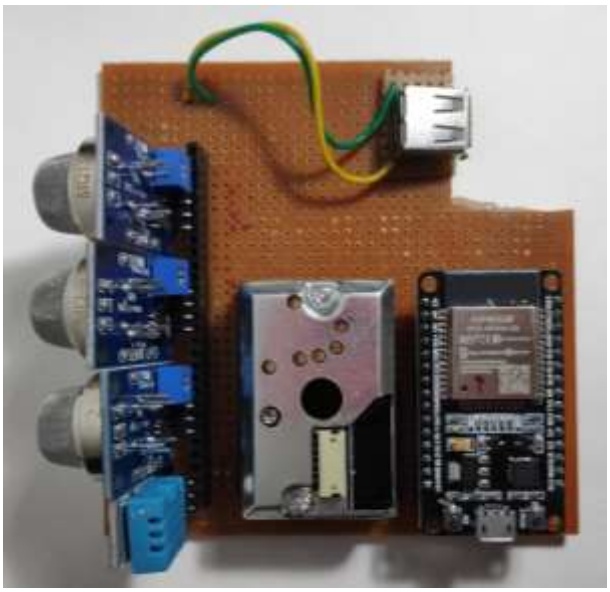
## 4.3 System Features:

- Cloud Networking
- MySQL Database
- Dashboard UI/UX
- Multiplicative Nodes
- Main Master Server with Database

## 4.4 External Interface Requirements:

### 4.4.1 User Interface:

### 4.4.2 Hardware Interface:



- MQ2
- MQ9
- MQ135
- DHT11
- GP2Y Dust Sensor
- NodeMCU ESP32
- Raspberry Pi 4

### 4.4.3 Software Interface:

- Python
- Php
- MySQL
- SQLlite
- HTML
- CSS
- MicroPython

### 4.4.4 Communications Interfaces:

- MySQL
- MQTT
- No-IP DNS routing

# 9. TEST CASES

| Sr.No. | Action | Inputs | Expected Output | Actual Output | Test Location | Test Result | Test Comments |
|---|---|---|---|---|---|---|---|
| 1. | Start Raspberry Pi | Power in | Raspberry Pi starts, SSH connection successful | Raspberry Pi starts, SSH connection successful | Putty Windows | Pass | Power On Successful |
| 2. | Start NodeMCU ESP32 (Node 1) | Power in | Node 1 Starts, connection successful with Master Server | Node 1 Starts, connection successful with Master Server | Serial Monitor – Arduino IDE | Pass | Connection with Master Server Successful |
| 3. | Connection Status | Serial Monitor (Debugging) | Connected to SSID (WiFi) | Connected to SSID (WiFi) | Serial Monitor – Arduino IDE | Pass | Connected to TestWifi |
| 4. | SQL Database | View Incoming Data | Data starts gathering into Database | Database gathers into the database with 5ms delay | Phpmyadmin On Google Chrome | Pass | Sensor Data Received |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 5. | Dashboard | User View | Sensor output shows different graph and % outputs | Sensor output shows different graph and % outputs | https://localhost - Google Chrome | Pass | Output Displayed |

# 10. CONCLUSION

The Paper Present An Raspberry Pi 4 And Nodemcu Esp32 Based Air Quality Monitoring System With Cloud Integration Shows The Humidity, Temperature And Smoke Graphs Over User Interface Also All Information Gathered By ,MQ2, MQ9, MQ135, DHT11, GP2Y Dust Sensor, Nodemcu ESP32, Raspberry Pi 4 These All Sensors And Hardware Stuff And We Get Accurate And Real-Time Information Of Humidity, Temperature & Smoke Etc. Also Used All Information To Monitor Air Quality In Surrounding.