IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

Low-Cost ESP32 Environmental Sensing Node with Cloud Alerting for Allergy-Relevant Conditions: A Documented Baseline

Monish S[1], Darius A[2], Rayhan Hameed[3]
[1],[2],[3] Department of Electronics and Communication (Advanced Communication Technology)
Chennai Institute of Technology (Autonomous), Chennai, Tamil Nadu, India
[1] Monishs.act2024@citchennai.net

Abstract: *We describe a low-cost ESP32 node that combines a BME280 (temperature, humidity, pressure) and an MQ135 gas sensor. It shows readings on a 16×2 LCD, uploads them every 30 s to the Blynk IoT cloud, and logs a cloud event when the raw gas reading crosses a threshold. The node screens for conditions associated with allergic respiratory symptoms through proxy variables; it does not detect or count pollen. In a four-site preliminary survey (home, park, traffic junction, construction site), the gas index was 59 to 90 at the two low-traffic sites and 270 to 290 at the two polluted ones. With one reading per site, only descriptive values are reported. We document the implementation for replication, state what proxy sensing can and cannot support, and give a validation protocol for a follow-up study.*

Keywords: ESP32, MQ135, BME280, Blynk, allergy risk, environmental proxies, low-cost sensing

I. INTRODUCTION

Pollen allergy affects a large share of urban populations, and direct pollen monitoring remains manual or expensive. Reference methods rely on volumetric spore traps read under a microscope, and automatic optical instruments are still costly [4]. Low-cost nodes are attractive because they can be deployed densely, but most published and hobbyist nodes report general air-quality values and say little about what those values mean for an allergic person.

Scope. The node described here does not detect or count pollen. It measures temperature, humidity, pressure and a broad-spectrum gas signal, and treats them as environmental context in which allergic symptoms are more or less likely. Any statement about pollen risk is therefore a statement about proxies and must be validated against a pollen reference before clinical use.

Contributions. (i) A specification of the reference implementation (pins, I²C addresses, thresholds, timing, cloud datastreams) from which another group can rebuild the node. (ii) A two-state air classification computed on the node and mirrored as a cloud event. (iii) A preliminary four-environment survey in which the gas channel separates low-traffic from polluted sites. (iv) An explicit statement of the validity limits of proxy-based allergy screening, and a statistical validation protocol for a follow-up study. We do not claim a new sensor, a new algorithm or a validated predictor. The contribution is a documented baseline with its limits stated.

II. LITERATURE REVIEW

Prior work falls into four groups (Table 1). The node belongs to the first group on hardware and differs from the others in what it measures.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40868

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

Group	Examples	Sensing	Direct pollen measurement	Relation to this work
Low-cost IoT air-quality nodes	Blynk-based Arduino system [9]; ESP32 classroom system with MQ135, DHT22 and Blynk [1]	MQ135 with Blynk on NodeMCU or ESP32; ppm readout and LED/buzzer alarm above 150 ppm in one [9]; DHT22 and LCD in the other, tested in simulation [1]	No	Same hardware class; this work adds an allergy-oriented scope statement and a replication-level specification
Wearable allergy monitors	Wearable monitor for dust-sensitive users [10]	PM1/2.5/10, NO_2, NH_3, temperature, humidity; threshold alerts	No	Richer particle sensing; same threshold-alert idea
Automatic pollen sensors	Beenose [4]	Multi-angle optical scattering with machine-learning classification; 9 of 12 species identified with scores above 78%	Yes	Different class; this node cannot identify pollen
Pollen proxies and forecasts	[3], [6], [8]	Low-cost optical particle counters with neural networks and random forests (Spearman 0.85, F1 up to 0.83) [8]; meteorology-based forecasts with accuracy above 80% up to a week ahead [3]; ANN with meteorology and pollutants [6]	Trained against reference pollen counts	Shows that meteorological variables carry pollen information when a pollen reference exists, which motivates the Validation Protocol in Section IV

Table 1: Related Work and its Relation to this Node

Three points follow. First, ESP32-class nodes with MQ135 and Blynk are common, so the hardware is not the contribution. Second, the studies that estimate pollen from low-cost sensors depend on a reference pollen series to train or validate against [3], [6], [8]; the present node has none, so it is a screening aid and not a pollen estimator. Third, the MQ135 is a broadly responsive sensor, not a selective CO_2 instrument, and a ppm conversion needs unit-specific calibration [7]. We therefore report raw ADC counts and a threshold state, not ppm.

III. METHODOLOGY

Hardware and Data Path:

The node reads two sensors, shows the values locally and mirrors them to the cloud (Figure 1, Table 2). The BME280 is read over I^2C [2]. The MQ135 module's analog output is read with analogRead on GPIO32, which belongs to the ESP32's ADC1 block and remains usable while Wi-Fi is active [5]. The result is a raw count (0 to 4095 at the default 12-bit resolution). Every 30 s a Blynk timer writes temperature, humidity and the raw gas count to virtual pins V0, V1 and V2, where a dashboard shows them as radial gauges.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

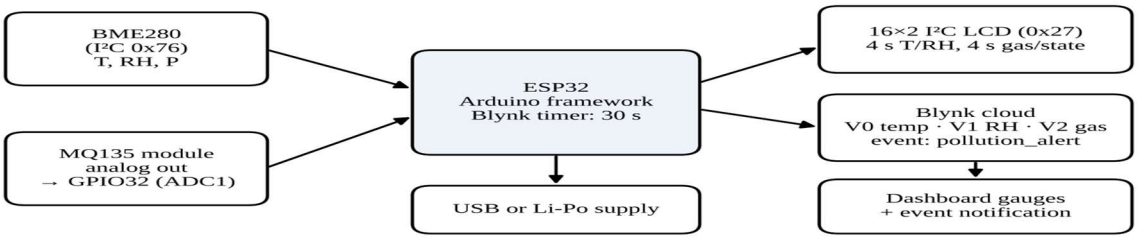


Figure 1: Block Diagram of the Node as Implemented (BME280 and 16×2 I²C LCD, as in the Firmware)

Element	Implementation
Controller	ESP32 (Arduino framework, Blynk ESP32 library)
Temperature, humidity, pressure	BME280, I²C address 0x76 (0x77 as alternative) [2]
Gas signal	MQ135 module analog output on GPIO32, raw ADC counts via analogRead [7]
Local display	16×2 I²C character LCD at address 0x27, custom degree glyph
Cloud datastreams	Blynk virtual pins V0 (temperature), V1 (humidity), V2 (gas value); dashboard range of V2 set to 0–600
Upload interval	30 s (Blynk timer)
Display cycle	Temperature and humidity for 4 s, then gas value and air state for 4 s
Air-state rule	Raw gas count below 1200: “Fresh Air”; 1200 or above: “Bad Air”
Cloud alert	Event pollution_alert logged by the node at each upload while the gas count is at or above 1200 (see Alert Rule)
Input check	Upload skipped when a BME280 reading is not a number

Table 2: Implementation Parameters

Alert Rule:

The threshold is applied to raw ADC counts so that the rule can be reproduced without a gas-calibration setup. The two-state rule is deliberately simple: it is transparent, cheap to compute, and avoids presenting an uncalibrated ppm figure as a measurement. The firmware listing retained from the project (Appendix A) tests the LCD state with “below 1200” and the cloud event with “above 1200”, which differ only at a count of exactly 1200. The reference implementation specified here uses one rule, count ≥ 1200, for both, and Appendix A is written that way. A cloud event can be forwarded by Blynk as an e-mail, push or SMS notification according to the event's configuration.

Sensor Handling:

The manufacturer's datasheet specifies a preheat of more than 24 h for the MQ135 before its readings are stable [7]. The project record does not state the preheat time, averaging or dwell time before each survey reading was taken, so we cannot confirm that this was observed, and the lowest readings in Section IV in particular should be treated as uncalibrated and indicative only. The module's analog output can approach its 5 V supply while the ESP32 ADC accepts at most 3.3 V, so a replication build should add a resistive divider; the record does not state whether the prototype had one. The replication specification therefore requires a ≥24 h preheat, a clean-air baseline reading, and a divider.

Known Implementation Constraints:

The firmware logs the alert event at every 30 s upload while the condition holds, with no hysteresis or rate limit. The display loop blocks for about 8 s per cycle through two fixed delays. Blynk start-up blocks until Wi-Fi connects, so the node does not display data when no network is present at boot. The prototype was built on a breadboard and powered from USB or a Li-Po cell.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

Survey Procedure:

The node was placed in four environments: indoors at home, a park, a traffic junction and a construction site. One reading per site was taken from the dashboard gauges. The duration of exposure before reading and the time of day were not recorded.

IV. RESULTS AND DISCUSSION

Survey Readings:

Table 3 reproduces the readings as recorded, one per site, in the units of the dashboard gauges (temperature and humidity 0–100, gas 0–600). The project's report labels the last numeric column "AQI"; we call it a gas index because it is not an air-quality index in any regulatory sense. The qualitative remarks are the authors' own impressions and are not outputs of the firmware rule.

Site	Temperature (gauge, 0–100)	Humidity (gauge, 0–100)	Gas index (gauge, 0–600)	Authors' remark
Indoor (home)	29	52	90	Good
Park	27	65	59	Good and clean
Traffic junction	35	62	290	Very poor
Construction site	37	49	270	Poor

Table 3: Readings in Four Environments, as Recorded (Dashboard Gauge Units)

The gas channel separates the two low-traffic sites (59 and 90) from the two polluted sites (270 and 290), while temperature and humidity vary less. Table 4 gives descriptive statistics. The values are un-normalised single readings, so no confidence intervals are given.

Variable	Mean	SD	Range
Temperature	32.0	4.8	27 to 37
Humidity	57.0	7.7	49 to 65
Gas index	177.3	119.6	59 to 290

Table 4: Descriptive Summary of Table 3 (n = 4 Sites, Gauge Units)

Rank association. Spearman's rank correlation between temperature and gas index is 0.80, and between humidity and gas index is −0.40. Ranks are unchanged by any monotonic rescaling, so these values do not depend on the unresolved gauge-to-physical-unit mapping discussed under Scales and the Alert Threshold. With four points the exact two-sided p-value for a coefficient of 0.80 is 0.33, and the smallest p-value any Spearman test can reach at n = 4 is 0.083. No association can therefore be declared significant, and the coefficients show only the direction of the pattern. A chi-square test is not applicable because it needs counts of independent observations across categories, and the data here are four single readings.

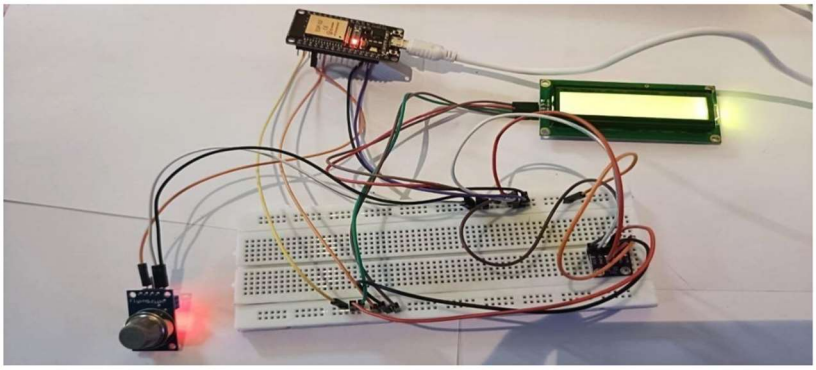
Scales and the Alert Threshold:

The firmware in Appendix A writes raw ADC counts to V2 and applies its threshold of 1200 to raw counts. The survey readings in Table 3, and the dashboard snapshots in Figure 2, lie on a 0–600 display scale: the V2 datastream is defined on 0–600, and the maximum survey value is 290. The retained firmware listing contains no step that converts counts to this scale, and the project record also contains a second threshold definition (an alert on the 0–600 scale at 400). The record does not show which firmware version was running when the survey readings and the alerts were taken. For orientation only, a linear mapping of the 12-bit range onto 0–600 would place a count of 1200 at about 176 on the display scale; we have not tested this and do not rely on it. The consequences are stated plainly: (a) the qualitative remarks in Table 3 were not computed by either threshold rule; (b) we cannot state which survey readings would or would not have raised an alert; (c) for temperature, the project record is inconsistent on whether the gauge shows °C or a normalised value, so Table 3 is not interpreted in physical units. For humidity the 0–100 gauge range coincides with %RH. Resolving

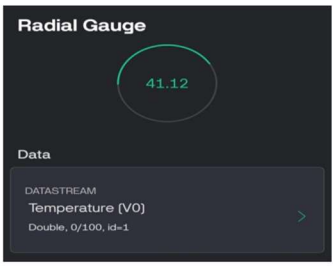
IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

these points is the first task of the follow-up study, in which the raw count, the displayed value and the alert state will be logged together.

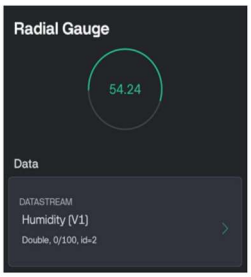
(a) Breadboard prototype (ESP32, BME280, MQ135 module, 16×2 LCD)



(b) V0 temperature (0–100)



(c) V1 humidity (0–100)



(d) V2 gas value (0–600)

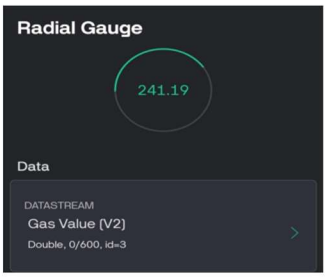


Figure 2: (a) Prototype. (b)–(d) Blynk gauges for the three datastreams. The panels were captured in separate experiments under different conditions; they show that all three channels reached the cloud and are not repeat measurements of one environment

Cloud and Alert Path:

The three datastreams reached the Blynk dashboard (Figure 2). The authors report receiving the pollution alert as e-mail and as SMS during bench testing. No screenshots or logs of the notifications were retained and the prototype has since been dismantled, so these are observations by the authors and not evidence presented here. Delivery latency, reliability and false-alarm rate were not measured. The data support one narrow claim: the node produced distinguishable gas readings in clean and polluted settings and delivered them to a dashboard. They do not support claims of accuracy, prediction or pollen detection.

Limitations:

1. No pollen measurement. Temperature and humidity are associated with pollen concentration in the forecasting literature [3], [6], [8], but they do not show that pollen is present.
2. Sensor selectivity and preparation. The MQ135 responds to many gases and to humidity, so the gas index is a general pollution signal. The threshold was not derived from a reference instrument, and preheat, averaging and baseline are undocumented (Sensor Handling, Section III).
3. Sample size. One reading per site, taken once, gives no estimate of variability, drift or repeatability.
4. Scale and firmware provenance. The mapping between raw counts and the 0–600 display scale, and the threshold in force during the survey, are not documented (Scales and the Alert Threshold, Section IV).
5. Alert path. Alerts are reported by the authors but were not logged or measured (Cloud and Alert Path, Section IV).
6. Single unit. Results come from one prototype, which has since been dismantled, so no measurement can be repeated on it.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40868

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

Validation Protocol for a Follow-up Study:

The aim is to test whether the node's air state tracks days of high pollen. The steps are a plan and no result is claimed for them.

1. Rebuild the node with the published parameters, add a ≥24 h preheat, a clean-air baseline and an input divider, and log the raw count, the displayed value, the alert state and a timestamp to the cloud for at least 6 weeks at two or more outdoor sites.
2. Obtain a daily or hourly pollen count (or, failing that, an AQI series) for the same area from a public station, and optionally a daily symptom log from consenting volunteers.
3. Classify each day as high or low pollen using the reference, and as alert or no alert using the node. Test association with a chi-square test of independence, using Fisher's exact test if any expected count is below 5.
4. Report agreement with Cohen's kappa, and sensitivity, specificity and ROC area for a sweep of gas thresholds in place of the single fixed 1200.
5. Measure monotonic association between the node's variables and the reference with Spearman's coefficient, and test site differences with the Kruskal–Wallis test.
6. Compare the gas channel with a reference sensor using regression and Bland–Altman limits of agreement.

V. CONCLUSION

This paper documents a low-cost ESP32 node that reports temperature, humidity and a raw gas signal locally and to the cloud, with a threshold-based air state mirrored as a cloud event. In four environments the gas channel separated low-traffic from polluted sites, but the sample is too small for statistical inference, the node does not measure pollen, and the scale and threshold in force during the survey cannot be fully reconstructed. The contribution is a documented baseline, a plain statement of what proxy sensing can support, and a validation protocol. Future work is to execute that protocol against a pollen or AQI reference, calibrate the gas channel, add a particle sensor, and replace the fixed threshold with one tuned on reference data.

ACKNOWLEDGMENT

The authors thank their project guide, Dr. K. Anand, the Head of the Department, Dr. K. Lakshmi Prabha, and Chennai Institute of Technology for their support.

DATA AND CODE AVAILABILITY

The survey readings are all given in Table 3; no further measurement data were retained. The firmware logic is listed in Appendix A with network and cloud credentials removed. The listing was repaired from a project report whose typesetting had damaged the original code, and it has not been recompiled because the hardware has been dismantled.

REFERENCES

[1]. Afendy, J. A., Hanif, M., & Gustin, A. P. N. (2026). Design of an Internet of Things (IoT)-based automatic classroom air quality monitoring and notification system. Antivirus: Jurnal Ilmiah Teknik Informatika, 20(1), 60-71. DOI: 10.35457/52nvvb47.

[2]. Bosch Sensortec. (n.d.). BME280: Combined humidity and pressure sensor. Datasheet BST-BME280-DS002.

[3]. Bulanda, D., Bulanda, M., Sacha, M., Horzyk, A., & Myszkowska, D. (2026). Comparison of machine learning methods in forecasting and characterizing the birch and grass pollen season. PLoS ONE, 21(2). DOI: 10.1371/journal.pone.0332093.

[4]. El Azari, H., Renard, J.-B., Lauthier, J., & Dudok de Wit, T. (2023). A laboratory evaluation of the new automated pollen sensor Beenose: Pollen discrimination using machine learning techniques. Sensors, 23(6), 2964. DOI: 10.3390/s23062964.

[5]. Espressif Systems. (n.d.). ESP32 series datasheet. Manufacturer datasheet.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, November 2026 Impact Factor: 8.2

[6]. Goudarzi, G., Birgani, Y. T., Assarehzadegan, M. A., Neisi, A., Dastoorpoor, M., Sorooshian, A., & Yazdani, M. (2022). Prediction of airborne pollen concentrations by artificial neural network and their relationship with meteorological parameters and air pollutants. Journal of Environmental Health Science and Engineering, 20(1), 251-264. DOI: 10.1007/s40201-021-00773-z.
[7]. Hanwei Electronics Co., Ltd. (n.d.). MQ-135 gas sensor: Technical data. Manufacturer datasheet.
[8]. Mills, S., Bousiotis, D., Maya-Manzano, J., Tummon, F., MacKenzie, A. R., & Pope, F. (2023). Constructing a pollen proxy from low-cost optical particle counter (OPC) data processed with neural networks and random forests. Science of the Total Environment, 161969. DOI: 10.1016/j.scitotenv.2023.161969.
[9]. Muhd Zain, N., Mohd Rozi, M. A. R., Othman, M., & Paidi, Z. (2024). The development of an IoT-based air quality monitoring system using the Blynk application. Journal of Computing Research and Innovation, 9(1), 157-166. DOI: 10.24191/jcrinn.v9i1.426.
[10]. Padmaloshani, P., Soundar Rajan, G., Dinesh, A., Vignesh, N., & Aananthaeswaran, M. (2026). Wearable environmental allergy monitoring and alert system for dust sensitive individuals. International Journal of Research in Applied Science and Engineering Technology (IJRASET), 14(4). DOI: 10.22214/ijraset.2026.81186.

APPENDIX A: REFERENCE FIRMWARE LOGIC

Libraries: WiFi, BlynkSimpleEsp32, Adafruit_Sensor, Adafruit_BME280, LiquidCrystal_I2C.

```
#define BLYNK_TEMPLATE_ID   "<your-template-id>"
#define BLYNK_TEMPLATE_NAME "Air Quality Monitoring"
#define BLYNK_AUTH_TOKEN    "<your-auth-token>"
#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#include <LiquidCrystal_I2C.h>
const int  GAS_PIN = 32;          // ADC1, raw counts 0..4095
const int  GAS_THRESHOLD = 1200;  // raw counts, applied as >=
char ssid[] = "<wifi-name>", pass[] = "<wifi-password>";
LiquidCrystal_I2C lcd(0x27, 16, 2);
Adafruit_BME280 bme;
BlynkTimer timer;

void sendSensor() {                    // every 30 s
  float t = bme.readTemperature(), h = bme.readHumidity();
  int gas = analogRead(GAS_PIN);
  if (isnan(t) || isnan(h)) return;       // input check
  Blynk.virtualWrite(V0, t);  Blynk.virtualWrite(V1, h);  Blynk.virtualWrite(V2, gas);
  if (gas >= GAS_THRESHOLD)
    Blynk.logEvent("pollution_alert", "Bad Air Quality Detected!");
}
void setup() {
  Serial.begin(115200);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);   // blocks until Wi-Fi connects
  if (!bme.begin(0x76)) while (1);            // try 0x77 if not detected
  lcd.begin();  /* custom degree glyph created here */
```

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, November 2026
Impact Factor: 8.2

```
  timer.setInterval(30000L, sendSensor);
}
void loop() {
  Blynk.run();  timer.run();
  float t = bme.readTemperature(), h = bme.readHumidity();
  int gas = analogRead(GAS_PIN);
  lcd.setCursor(0,0); lcd.print("Temp: "); lcd.print(t); /* degree glyph */ lcd.print("C");
  lcd.setCursor(0,1); lcd.print("Hum: ");  lcd.print(h); lcd.print("%");
  delay(4000); lcd.clear();
  lcd.setCursor(0,0); lcd.print("Gas: "); lcd.print(gas);
  lcd.setCursor(0,1); lcd.print(gas < GAS_THRESHOLD ? "Fresh Air :)" : "Bad Air !!!");
  delay(4000); lcd.clear();
}
```