

## PERANCANGAN PROTOTIPE ALAT *INTERNET OF THINGS* (IOT) UNTUK PEMANTAUAN KONDISI LINGKUNGAN PADA RUANG PENYIMPANAN BAWANG MERAH MENGGUNAKAN ESP32, MQ-135 DAN DHT22

Indra Aditya Purba<sup>1</sup>, Firahmi Rizky<sup>2</sup>

<sup>1,2</sup>Universitas Muhammadiyah Sumatera Utara, Indonesia

Email: [indraaditya90@gmail.com](mailto:indraaditya90@gmail.com)<sup>1</sup>, [firahmirizky@umsu.ac.id](mailto:firahmirizky@umsu.ac.id)<sup>2</sup>

### ABSTRAK

Bawang merah merupakan komoditas hortikultura yang rentan mengalami penurunan mutu selama penyimpanan akibat perubahan suhu, kelembapan, dan kualitas udara. Pemantauan secara manual memiliki keterbatasan dalam menyediakan informasi kondisi lingkungan secara kontinu dan *real-time*. Penelitian ini bertujuan merancang dan mengimplementasikan prototipe *Internet of Things* (IoT) untuk memantau kondisi lingkungan ruang penyimpanan bawang merah menggunakan ESP32, sensor DHT22, dan MQ-135 yang terintegrasi dengan *Arduino IoT Cloud*. Penelitian menggunakan metode *prototyping* yang meliputi analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi sistem. Sensor DHT22 digunakan untuk mengukur suhu dan kelembapan, sedangkan MQ-135 digunakan untuk mendeteksi perubahan kualitas udara berdasarkan nilai *Analog to Digital Converter* (ADC). Data sensor diproses oleh ESP32 dan dikirimkan melalui jaringan Wi-Fi ke *Arduino IoT Cloud* untuk ditampilkan pada *dashboard* berbasis web dan Android serta digunakan sebagai dasar klasifikasi kondisi ruangan. Hasil pengujian menunjukkan bahwa sistem mampu membaca dan mengirimkan data sensor secara kontinu dengan interval 2 detik. Sensor DHT22 menghasilkan pembacaan suhu pada rentang 29,6–30,1 °C dan kelembapan 71,2–71,9% RH, sedangkan sensor MQ-135 menghasilkan nilai 234–421 ADC dan dapat digunakan sebagai indikator relatif perubahan kualitas udara. Sistem juga berhasil menampilkan data secara *real-time*, mempertahankan koneksi ESP32 dengan *Arduino IoT Cloud*, serta mengirimkan notifikasi ketika kondisi ruangan mencapai kategori bahaya. Dengan demikian, prototipe yang dikembangkan dapat digunakan sebagai sistem pemantauan dan peringatan dini terhadap perubahan kondisi lingkungan pada ruang penyimpanan bawang merah secara jarak jauh. Kata kunci: *Internet of Things*, ESP32, MQ-135, DHT22, bawang merah, *monitoring*, *Arduino IoT Cloud*

### ABSTRACT

Shallots are horticultural commodities susceptible to quality degradation during storage due to fluctuations in temperature, humidity, and air quality. Manual monitoring has limitations in providing continuous, real-time information regarding environmental conditions. This study aims to design and implement an *Internet of Things* (IoT) prototype to monitor the environmental conditions of a shallot storage room using an ESP32, DHT22 sensor, and MQ-135 sensor integrated with the *Arduino IoT Cloud*. The research employs a *prototyping* method comprising requirements analysis, design, implementation, testing, and system evaluation. The DHT22 sensor measures temperature and humidity, while the MQ-135 sensor detects air quality changes based on *Analog-to-Digital Converter* (ADC) values. Sensor data is processed by the ESP32 and transmitted via Wi-Fi to the *Arduino IoT Cloud* for display on web- and Android-based dashboards, as well as for use in classifying room conditions. Test results demonstrate that the system is capable of continuously reading and transmitting sensor data at 2-second intervals. The DHT22 sensor yielded temperature readings ranging from 29.6–30.1 °C and humidity levels of 71.2–71.9% RH, while the MQ-135 sensor produced ADC values between 234 and 421, serving as a relative indicator of air quality changes. The system successfully displayed real-time data, maintained the ESP32 connection to the *Arduino IoT Cloud*, and sent notifications when room conditions reached a hazardous level.

*Thus, the developed prototype can serve as a remote monitoring and early warning system for environmental changes in shallot storage facilities.*

*Keywords: Internet of Things, ESP32, MQ-135, DHT22, shallots, monitoring, Arduino IoT Cloud*

## PENDAHULUAN

Bawang merah (*Allium cepa* L.) merupakan salah satu komoditas hortikultura strategis di Indonesia yang memiliki peran penting dalam stabilitas pasokan dan harga bahan pangan. Berdasarkan data Badan Pusat Statistik (BPS, 2024), produksi bawang merah nasional mencapai lebih dari 1,9 juta ton per tahun. Dalam rantai distribusinya, komoditas ini umumnya melalui tahap penyimpanan sebelum dipasarkan. Pada fase pasca panen (*post-harvest*), bawang termasuk komoditas yang rentan mengalami penurunan mutu akibat faktor lingkungan penyimpanan yang kurang terkontrol.

Secara global, kerugian pasca panen pada komoditas hortikultura dilaporkan berada pada kisaran 20–40% tergantung pada sistem penanganan dan infrastruktur penyimpanan (FAO, 2019; Kitinoja & Saran, 2020). Di Indonesia, kehilangan hasil bawang merah selama proses penyimpanan dilaporkan dapat mencapai 15–30%, yang sebagian besar dipengaruhi oleh kondisi lingkungan yang tidak sesuai serta serangan mikroorganisme selama masa simpan (Direktorat Jenderal Hortikultura, 2022). Kerugian tersebut berdampak langsung pada nilai ekonomi komoditas dan efisiensi distribusi.

Secara fisiologis, bawang yang telah dipanen masih melakukan respirasi dan metabolisme. Suhu dan kelembapan relatif (*Relative Humidity/RH*) menjadi faktor utama yang memengaruhi laju respirasi, pertumbuhan mikroorganisme, serta daya simpan umbi. Kondisi kelembapan yang terlalu tinggi dapat meningkatkan risiko pertumbuhan jamur patogen seperti *Aspergillus spp.* dan *Fusarium spp.*, yang merupakan penyebab umum pembusukan pada bawang selama penyimpanan (Kader, 2002; Thompson et al., 2008). Oleh karena itu, kestabilan kondisi lingkungan ruang penyimpanan menjadi faktor penting dalam menjaga kualitas komoditas.

Selain perubahan suhu dan kelembapan, proses degradasi biologis pada komoditas hortikultura umumnya diikuti oleh perubahan kualitas udara di ruang penyimpanan. Aktivitas metabolisme dan mikroorganisme dapat menghasilkan senyawa volatil atau *Volatile organic compounds (VOC)* yang menyebabkan perubahan karakteristik udara dalam ruang tertutup (Baietto & Wilson, 2015; Kim et al., 2020). Peningkatan konsentrasi gas volatil tersebut tidak selalu langsung terlihat melalui perubahan fisik komoditas, namun dapat menjadi indikator adanya perubahan kondisi lingkungan yang berpotensi mempercepat penurunan mutu.

Pada praktik konvensional, pemantauan kondisi ruang penyimpanan bawang masih dilakukan melalui inspeksi visual serta pengukuran suhu dan kelembapan secara manual. Pendekatan ini memiliki keterbatasan karena tidak memberikan data secara kontinu dan *real-time* serta bergantung pada subjektivitas pengamatan. Sementara itu, pembusukan bawang bersifat progresif; satu umbi yang mengalami

degradasi dapat memengaruhi kondisi mikro-lingkungan di sekitarnya, seperti peningkatan kelembapan lokal atau perubahan kualitas udara, yang pada akhirnya dapat memperbesar risiko kerusakan pada umbi lainnya apabila tidak terdeteksi sejak dini.

Perkembangan teknologi *Internet of Things (IoT)* memungkinkan integrasi berbagai sensor dengan sistem *monitoring* berbasis jaringan yang mampu mengirimkan data secara kontinu dan memberikan notifikasi ketika parameter tertentu melewati ambang batas. Salah satu mikrokontroler yang banyak digunakan dalam pengembangan sistem *IoT* adalah *ESP32* karena telah dilengkapi dengan konektivitas *Wi-Fi* dan *Bluetooth*, konsumsi daya yang relatif rendah, serta kemudahan integrasi dengan berbagai sensor dan platform aplikasi (Maier et al., 2017). Sensor *MQ-135* dikenal sensitif terhadap berbagai gas seperti amonia ( $NH_3$ ), karbon dioksida ( $CO_2$ ), serta beberapa senyawa *VOC* lainnya (Hanwei Electronics, 2019). Dalam penelitian ini, sensor tersebut tidak digunakan untuk mengidentifikasi jenis gas secara spesifik, melainkan untuk mendeteksi perubahan konsentrasi gas volatil secara relatif sebagai indikator perubahan kualitas udara dalam ruang penyimpanan. Selain itu, sensor *DHT22* digunakan untuk memantau suhu dan kelembapan relatif yang berperan penting dalam menjaga stabilitas kondisi penyimpanan (Aosong Electronic, 2020).

Beberapa penelitian sebelumnya telah membahas pengendalian suhu dan kelembapan untuk memperpanjang masa simpan komoditas hortikultura. Namun, sebagian besar penelitian berfokus pada sistem skala industri dengan biaya implementasi relatif tinggi atau menggunakan sensor spesifik yang kurang aplikatif untuk skala kecil hingga menengah (Baietto & Wilson, 2015; Kim et al., 2020). Integrasi antara pemantauan gas volatil dan parameter lingkungan dalam satu sistem *monitoring* berbasis *IoT* yang sederhana, terjangkau, dan ditujukan sebagai sistem peringatan dini pada ruang penyimpanan bawang masih terbatas.

Dalam konteks penyimpanan bawang, pembusukan yang terjadi pada satu atau beberapa umbi dapat meningkatkan kelembapan lokal serta memengaruhi kualitas udara di dalam ruang penyimpanan. Perubahan kondisi tersebut berpotensi mempercepat degradasi komoditas lain di sekitarnya apabila tidak terdeteksi sejak dini. Oleh karena itu, diperlukan suatu sistem yang mampu melakukan pemantauan parameter lingkungan secara kontinu sebagai dasar pengambilan keputusan untuk mencegah meluasnya kerusakan.

Sistem yang dirancang dalam penelitian ini tidak ditujukan untuk menganalisis komposisi kimia gas secara spesifik, melainkan untuk memonitor perubahan parameter lingkungan terhadap kondisi awal sebagai bentuk sistem peringatan dini. Informasi yang diperoleh melalui *Arduino Cloud* dapat digunakan sebagai dasar tindakan korektif, seperti peningkatan ventilasi atau pemisahan komoditas yang terindikasi mengalami penurunan kualitas. Dengan demikian, sistem ini berfungsi sebagai pendukung dalam upaya pencegahan penyebaran pembusukan pada ruang

penyimpanan bawang melalui *monitoring* kualitas udara dan kondisi lingkungan secara terintegrasi.

Berdasarkan uraian tersebut, penelitian ini dilakukan untuk merancang dan membangun prototipe alat *Internet of Things* (IoT) yang mampu melakukan *monitoring* kualitas udara dan kondisi lingkungan sebagai upaya pencegahan penyebaran pembusukan pada ruang penyimpanan bawang menggunakan mikrokontroler ESP32, sensor MQ-135, sensor DHT22.

### METODE PENELITIAN

Penelitian ini menggunakan metode *prototyping* untuk merancang dan mengembangkan sistem pemantauan kualitas udara pada ruang penyimpanan bawang merah berbasis *Internet of Things* (IoT). Metode *prototyping* dipilih karena memungkinkan proses pengembangan sistem dilakukan secara bertahap melalui pembuatan model awal, pengujian, evaluasi, dan penyempurnaan hingga sistem dapat berfungsi sesuai kebutuhan. Prototipe yang dikembangkan menggunakan mikrokontroler ESP32 sebagai pusat pengendali, sensor MQ-135 untuk mendeteksi gas volatil (*volatile organic compounds/VOC*), sensor DHT22 untuk mengukur suhu dan kelembapan, serta Arduino Cloud sebagai media *monitoring* yang memungkinkan data dipantau secara *real-time* melalui perangkat *smartphone*. Tahapan pengembangan meliputi analisis kebutuhan, perancangan sistem, pembuatan dan implementasi prototipe, serta pengujian dan evaluasi sistem.

Pengumpulan data dilakukan melalui observasi, studi pustaka, dan eksperimen. Observasi dilakukan dengan mengamati kondisi ruang penyimpanan bawang merah, terutama suhu, kelembapan, sirkulasi udara, serta perubahan fisik yang menunjukkan penurunan kualitas atau pembusukan. Studi pustaka dilakukan dengan menelaah buku, jurnal ilmiah, prosiding, penelitian terdahulu, dan dokumentasi teknis yang relevan dengan IoT, ESP32, sensor MQ-135, sensor DHT22, gas volatil, serta Arduino Cloud. Sementara itu, eksperimen dilakukan melalui proses perancangan, implementasi, dan pengujian prototipe untuk mengetahui kinerja sensor dalam membaca gas volatil, suhu, dan kelembapan, sekaligus menguji komunikasi data antara ESP32 dan Arduino Cloud serta mekanisme notifikasi sistem.

Pengembangan perangkat keras dilakukan dengan mengintegrasikan ESP32, sensor MQ-135, sensor DHT22, *power supply*, serta perangkat pendukung lainnya. ESP32 berfungsi menerima dan mengolah data sensor kemudian mengirimkannya melalui jaringan Wi-Fi ke Arduino Cloud. Sensor MQ-135 digunakan untuk memperoleh data mengenai gas volatil, sedangkan DHT22 digunakan untuk mengukur suhu dan kelembapan ruang penyimpanan. Data dari kedua sensor diproses oleh ESP32 dan dikirimkan ke Arduino Cloud untuk ditampilkan secara *real-time*. Pengembangan perangkat lunak dilakukan menggunakan Arduino IDE dengan dukungan *library* sensor MQ-135 dan DHT, sedangkan Fritzing digunakan dalam perancangan rangkaian elektronik dan *draw.io* untuk penyusunan *flowchart* sistem.

Proses kerja sistem dimulai dari inisialisasi ESP32, sensor MQ-135 dan DHT22, serta koneksi Wi-Fi ke Arduino Cloud. Sensor kemudian membaca gas volatil, suhu, dan kelembapan secara periodik. Data hasil pembacaan diproses dan dibandingkan dengan nilai *threshold* yang telah ditentukan untuk mengklasifikasikan kondisi ruang penyimpanan ke dalam kategori aman, waspada, atau bahaya. Parameter yang digunakan meliputi suhu 25–35 °C, kelembapan 50–70% RH, dan kualitas udara hingga 350 ADC sebagai kondisi aman. Data hasil pengukuran selanjutnya dikirim ke Arduino Cloud dan ditampilkan dalam bentuk nilai maupun grafik secara *real-time*. Apabila parameter berada di luar batas yang ditentukan, sistem memberikan notifikasi peringatan kepada pengguna. Seluruh proses berlangsung secara berulang selama sistem aktif sehingga memungkinkan pemantauan kondisi penyimpanan secara kontinu.

Data hasil eksperimen dianalisis berdasarkan kemampuan sistem dalam membaca parameter lingkungan, kestabilan pengiriman data, respons terhadap perubahan kondisi, serta kemampuan sistem memberikan peringatan ketika parameter melewati batas yang ditentukan. Evaluasi dilakukan untuk mengetahui kesesuaian antara fungsi sistem dengan rancangan yang telah ditetapkan. Apabila ditemukan ketidaksesuaian atau kekurangan selama pengujian, dilakukan penyempurnaan terhadap prototipe hingga sistem dapat berfungsi secara optimal dalam melakukan pemantauan kondisi ruang penyimpanan bawang merah.

## HASIL DAN PEMBAHASAN

### 1. Hasil Penelitian

#### a. Hasil Perancangan Alat

Berdasarkan tahapan perancangan yang telah dijelaskan pada Bab III, penelitian ini berhasil merealisasikan sebuah prototipe alat *Internet of Things (IoT)* untuk pemantauan kualitas udara dan kondisi lingkungan pada ruang penyimpanan bawang merah. Prototipe dikembangkan menggunakan mikrokontroler *ESP32 Dev Module* sebagai pusat pengendali sistem yang terintegrasi dengan sensor *DHT22* untuk mengukur suhu dan kelembapan serta sensor *MQ-135* untuk mendeteksi perubahan kualitas udara di dalam ruang penyimpanan.

Sistem dirancang agar mampu melakukan pembacaan data sensor secara berkala setiap 2 detik, kemudian mengirimkan hasil pengukuran ke platform *Arduino IoT Cloud* melalui jaringan *WiFi*. Data yang diterima selanjutnya ditampilkan secara *real-time* pada *Dashboard* berbasis web maupun aplikasi Android sehingga pengguna dapat memantau kondisi ruang penyimpanan dari jarak jauh. Selain itu, prototipe juga dilengkapi LED indikator sebagai penanda bahwa sistem beroperasi dengan baik serta fitur notifikasi ketika kondisi lingkungan mencapai status peringatan.



**Gambar 1 Tampak depan prototipe alat**

Gambar 1 menunjukkan bentuk keseluruhan prototipe yang telah direalisasikan. Seluruh komponen ditempatkan di dalam ruang penyimpanan sehingga sensor *DHT22* dan *MQ-135* dapat mendeteksi kondisi lingkungan secara langsung. Prototipe ini menjadi media utama dalam proses pengujian dan pengambilan data penelitian.



**Gambar 2 Tampak dalam rangkaian komponen**

Gambar 2 memperlihatkan susunan komponen elektronik yang terpasang pada prototipe. Penempatan *ESP32*, sensor *DHT22*, sensor *MQ-135*, serta LED indikator disesuaikan dengan rancangan yang telah dibuat sehingga memudahkan proses pembacaan sensor, komunikasi data, dan pemeliharaan perangkat.

#### **b. Impelementasi Kode Program**

Implementasi kode program dilakukan menggunakan *Arduino IDE 2.3.10* dengan bahasa pemrograman C++. Program dirancang untuk menginisialisasi

perangkat, membaca data sensor *DHT22* dan *MQ-135*, mengirimkan data ke *Arduino IoT Cloud*, menampilkan status sistem, serta mengaktifkan LED indikator sebagai penanda proses *monitoring*.

**Tabel 1 Keterangan Script Implementasi Program**

| No | Keterangan                                                                                                                                   | Script                                                                                                                                  |
|----|----------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Memasukkan file konfigurasi <i>Arduino Cloud</i> , <i>WiFi</i> , waktu, dan sensor <i>DHT22</i> .                                            | <code>#include "Arduino_secrets.h" #include "thingProperties.h" #include &lt;WiFi.h&gt; #include &lt;time.h&gt; #include "DHT.h"</code> |
| 2  | Menginisialisasi pin <i>DHT22</i> , <i>MQ-135</i> , LED indikator, serta interval pembacaan data 2 detik.                                    | <code>#define DHTPIN 4 #define MQ135_PIN 34 #define LED_PIN 2 const unsigned long interval = 2000;</code>                               |
| 3  | Menginisialisasi komunikasi serial, sensor, <i>Arduino Cloud</i> , koneksi <i>WiFi</i> , dan sinkronisasi waktu pada fungsi <i>setup()</i> . | <code>Serial.begin(115200); dht.begin(); initProperties(); ArduinoCloud.begin(...); configTime(...);</code>                             |
| 4  | Melakukan pembacaan suhu, kelembapan, dan kadar gas setiap 2 detik pada fungsi <i>loop()</i> .                                               | <code>dht.readTemperature(); dht.readHumidity(); analogRead(MQ135_PIN);</code>                                                          |
| 5  | Mengirim data sensor ke <i>Arduino Cloud</i> serta memperbarui status <i>ESP32</i> dan waktu pembaruan.                                      | <code>suhu = t; kelembapan = h; kadarGas = gas; statusESP32 = true; UpdateTerakhir = getJamWIB();</code>                                |
| 6  | Menentukan status ruangan, mengaktifkan variabel notifikasi, dan mengendalikan LED indikator.                                                | <code>statusRuangan = "AMAN"; peringatan = true; digitalWrite(LED_PIN, HIGH);</code>                                                    |

Berdasarkan Tabel 1, program diawali dengan konfigurasi *library* dan pendefinisian pin yang digunakan oleh *ESP32* sebagai pengendali sistem. Selanjutnya, pada fungsi *setup()* dilakukan inisialisasi komunikasi serial, sensor *DHT22*, *Arduino IoT Cloud*, koneksi *WiFi*, serta sinkronisasi waktu menggunakan *Network Time Protocol (NTP)* sehingga sistem mampu menampilkan waktu pembaruan dalam format WIB.

Proses utama sistem dijalankan pada fungsi *loop()* dengan metode pembacaan berkala setiap 2 detik menggunakan fungsi *millis()*. Data suhu, kelembapan, dan kualitas udara yang diperoleh dari sensor kemudian dikirim ke *Arduino IoT Cloud* untuk ditampilkan secara *real-time* pada *Dashboard*. Selain itu, program juga menentukan status kondisi ruangan berdasarkan parameter lingkungan, mengaktifkan variabel notifikasi ketika kondisi mencapai status bahaya, serta mengendalikan LED indikator sebagai penanda bahwa proses pembacaan dan pengiriman data sedang berlangsung.

### c. Implementasi Dashboard Arduino IoT Cloud

*Dashboard Arduino IoT Cloud* diimplementasikan sebagai antarmuka utama untuk menampilkan hasil pemantauan suhu, kelembapan, kualitas udara, status perangkat, dan waktu pembaruan secara *real-time*. *Dashboard* dapat diakses melalui web browser maupun aplikasi *Arduino Cloud Remote* pada perangkat Android, sehingga pengguna dapat memantau kondisi ruang penyimpanan bawang merah dari jarak jauh selama *ESP32* terhubung ke jaringan internet.

Pada penelitian ini, *Dashboard* dirancang dengan prinsip kemudahan pemantauan (*user-friendly*) dengan menampilkan informasi utama pada halaman awal. Setiap data yang dibaca oleh sensor *DHT22* dan *MQ-135* dikirimkan ke *Arduino IoT Cloud* setiap 2 detik, kemudian diperbarui secara otomatis pada seluruh widget yang telah dikonfigurasi.

**Tabel 2 Implementasi Widget Dashboard Arduino IoT Cloud**

| No | Widget                   | Fungsi                                                                         |
|----|--------------------------|--------------------------------------------------------------------------------|
| 1  | Header Monitoring System | Menampilkan identitas sistem <i>monitoring</i> ruang penyimpanan bawang merah. |
| 2  | Status Ruangan           | Menampilkan kondisi ruangan berdasarkan hasil pengolahan data sensor.          |
| 3  | Status <i>ESP32</i>      | Menunjukkan kondisi perangkat dalam keadaan Online atau Offline.               |
| 4  | <i>Update</i> Terakhir   | Menampilkan waktu pembaruan data terbaru dalam format WIB.                     |
| 5  | <i>Gauge</i> Suhu        | Menampilkan nilai suhu (°C) secara <i>real-time</i> .                          |
| 6  | <i>Gauge</i> Kelembapan  | Menampilkan nilai kelembapan (%) secara <i>real-time</i> .                     |
| 7  | <i>Gauge</i> Kadar Gas   | Menampilkan nilai pembacaan sensor <i>MQ-135</i> (ADC).                        |
| 8  | <i>Advanced Chart</i>    | Menampilkan grafik historis perubahan suhu, kelembapan, dan kadar gas.         |

Berdasarkan Tabel 2, *Dashboard* terdiri dari delapan widget yang saling terintegrasi dengan variabel pada *Arduino IoT Cloud*. Widget *Gauge* digunakan untuk memvisualisasikan nilai sensor secara langsung, sedangkan *Advanced Chart* berfungsi menampilkan riwayat perubahan data sehingga memudahkan pengguna melakukan analisis kondisi lingkungan selama proses penyimpanan bawang merah.

Selain itu, widget Status *ESP32* dan *Update Terakhir* digunakan sebagai indikator bahwa perangkat masih terhubung dengan *Arduino IoT Cloud* dan proses pengiriman data berlangsung secara normal. Dengan demikian, *Dashboard* tidak hanya berfungsi sebagai media visualisasi data, tetapi juga sebagai antarmuka pemantauan kondisi sistem secara menyeluruh.

## 2. Pengujian Seluruh Komponen Sistem

### a. Pengujian Koneksi *ESP32* dan *Arduino IoT Cloud*

Pengujian koneksi dilakukan untuk mengetahui kemampuan *ESP32* dalam terhubung dengan jaringan *WiFi* dan *Arduino IoT Cloud* sehingga proses pengiriman data sensor dapat berlangsung secara *real-time*. Pengujian ini dilakukan dengan menghubungkan *ESP32* ke jaringan internet, kemudian mengamati status koneksi melalui Serial Monitor dan *Dashboard Arduino IoT Cloud*.

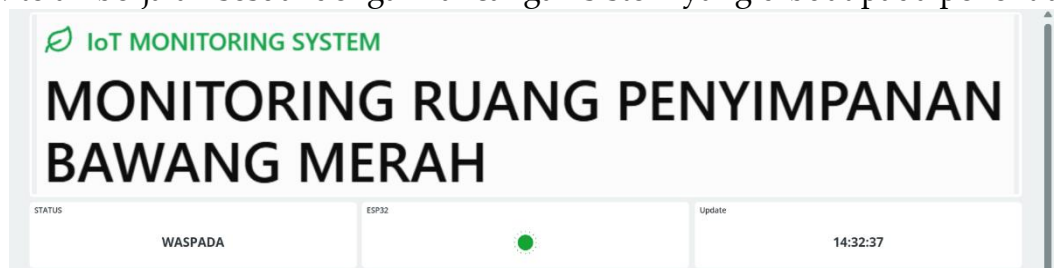
Keberhasilan pengujian ditandai dengan status perangkat Online, proses sinkronisasi waktu (*Network Time Protocol*), serta perubahan nilai variabel pada *Dashboard* setiap dua detik. Hasil pengujian menunjukkan bahwa *ESP32* berhasil mempertahankan koneksi dengan *Arduino IoT Cloud* selama proses *monitoring* berlangsung.

**Tabel 3 Hasil Pengujian Koneksi *ESP32* dan *Arduino IoT Cloud***

| No | Parameter Pengujian                                | Hasil    |
|----|----------------------------------------------------|----------|
| 1  | <i>ESP32</i> terhubung ke jaringan <i>WiFi</i>     | Berhasil |
| 2  | <i>ESP32</i> terhubung ke <i>Arduino IoT Cloud</i> | Berhasil |
| 3  | Status perangkat pada <i>Dashboard</i>             | Online   |
| 4  | Sinkronisasi waktu (WIB)                           | Berhasil |
| 5  | Interval pembaruan data                            | 2 detik  |

Berdasarkan Tabel 4.3, seluruh parameter koneksi menunjukkan hasil berhasil. *ESP32* mampu melakukan autentikasi ke *Arduino IoT Cloud* menggunakan *device key* yang telah dikonfigurasi pada *thingProperties.h*, kemudian mempertahankan komunikasi dengan server melalui jaringan *WiFi*. Status perangkat pada *Dashboard* berubah menjadi Online, sedangkan waktu pembaruan (*Update Terakhir*) ditampilkan sesuai zona waktu WIB hasil sinkronisasi *NTP*.

Selain itu, pengamatan pada *Dashboard* menunjukkan bahwa nilai suhu, kelembapan, dan kadar gas diperbarui secara otomatis setiap dua detik. Hal tersebut membuktikan bahwa komunikasi antara perangkat keras dan platform *Arduino IoT Cloud* telah berjalan sesuai dengan rancangan sistem yang dibuat pada penelitian ini.



**Gambar 3 Dashboard *Arduino IoT Cloud* dengan status Online.**

#### **b. Pengujian Sensor Suhu dan Kelembapan *DHT22***

Pengujian sensor *DHT22* dilakukan untuk mengetahui kemampuan sensor dalam membaca parameter suhu dan kelembapan pada ruang penyimpanan bawang merah serta mengirimkan hasil pembacaan ke *Arduino IoT Cloud* secara *real-time*.

Pengujian dilakukan dengan menempatkan sensor *DHT22* di dalam prototipe ruang penyimpanan bawang merah. Sistem dijalankan selama proses *monitoring*

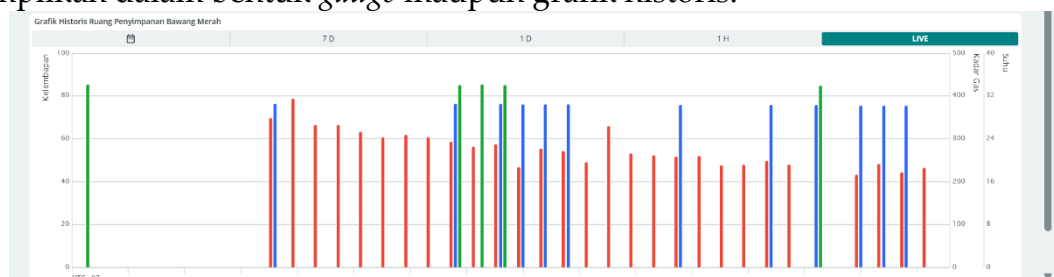
dengan interval pembacaan setiap 2 detik. Data suhu dan kelembapan diamati melalui Serial Monitor serta *Dashboard Arduino IoT Cloud*, kemudian dicatat sebanyak 10 kali pembacaan berturut-turut.

**Tabel 4 Hasil Pengujian Sensor DHT22**

| No | Waktu    | Suhu (°C) | Kelembapan (%RH) |
|----|----------|-----------|------------------|
| 1  | 23:00:02 | 29.6      | 71.2             |
| 2  | 23:00:04 | 29.7      | 71.3             |
| 3  | 23:00:06 | 29.7      | 71.4             |
| 4  | 23:00:08 | 29.8      | 71.5             |
| 5  | 23:00:10 | 29.8      | 71.5             |
| 6  | 23:00:12 | 29.9      | 71.6             |
| 7  | 23:00:14 | 29.9      | 71.6             |
| 8  | 23:00:16 | 30.0      | 71.7             |
| 9  | 23:00:18 | 30.0      | 71.8             |
| 10 | 23:00:20 | 30.1      | 71.9             |

Berdasarkan hasil pengujian pada Tabel 4, sensor *DHT22* mampu melakukan pembacaan suhu dan kelembapan secara kontinu dengan interval pembaruan setiap 2 detik. Nilai yang ditampilkan pada Serial Monitor dan *Arduino IoT Cloud* berubah secara *real-time* sesuai kondisi lingkungan di dalam ruang penyimpanan.

Selama pengujian, data yang diperoleh menunjukkan perubahan nilai yang relatif stabil sehingga menandakan bahwa sensor *DHT22* dapat berfungsi dengan baik sebagai perangkat pemantau kondisi lingkungan. Hasil pembacaan tersebut kemudian dikirimkan secara otomatis ke *Dashboard Arduino IoT Cloud* untuk ditampilkan dalam bentuk *gauge* maupun grafik historis.



**Gambar 4 Grafik suhu dan kelembapan pada *Arduino IoT Cloud*.**

### c. Pengujian Sensor Gas MQ-135

Pengujian sensor *MQ-135* dilakukan untuk mengetahui kemampuan sensor dalam mendeteksi perubahan kualitas udara pada ruang penyimpanan bawang merah. Nilai yang dihasilkan berupa data analog (*Analog to Digital Converter / ADC*) yang kemudian dikirimkan ke *Arduino IoT Cloud* dan ditampilkan secara *real-time* pada *Dashboard*.

Sensor *MQ-135* ditempatkan di dalam ruang penyimpanan bersama objek pengujian. Sistem dijalankan dengan interval pembacaan setiap 2 detik, kemudian

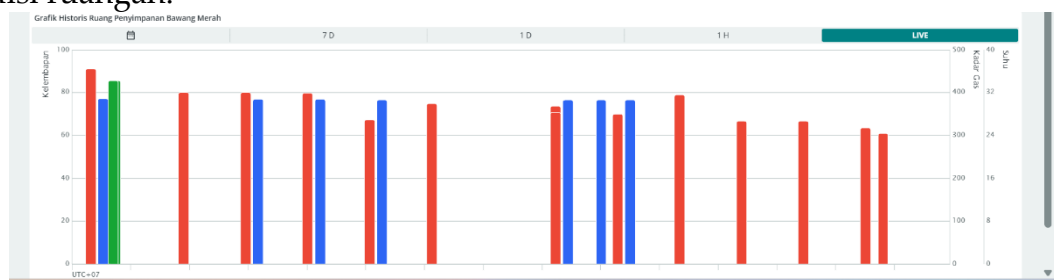
nilai ADC diamati melalui *Dashboard Arduino IoT Cloud*. Data hasil pembacaan dicatat sebanyak 10 kali sebagai sampel pengujian.

**Tabel 5 Hasil Pengujian Sensor MQ-135**

| No | Waktu Pengamatan | Nilai ADC | Status  |
|----|------------------|-----------|---------|
| 1  | 14:15            | 421       | WASPADA |
| 2  | 14:17            | 375       | WASPADA |
| 3  | 14:19            | 418       | WASPADA |
| 4  | 14:21            | 420       | WASPADA |
| 5  | 14:23            | 374       | WASPADA |
| 6  | 14:25            | 373       | WASPADA |
| 7  | 14:27            | 372       | WASPADA |
| 8  | 14:29            | 371       | WASPADA |
| 9  | 14:31            | 420       | WASPADA |
| 10 | 14:33            | 234       | AMAN    |

Berdasarkan Tabel 5, sensor MQ-135 menghasilkan nilai pembacaan pada rentang 234–421 ADC selama proses pengujian. Sebagian besar data berada pada kisaran 351–550 ADC sehingga sistem mengklasifikasikan kondisi ruang penyimpanan sebagai WASPADA berdasarkan parameter yang telah ditetapkan pada Bab III. Pada akhir pengamatan, nilai sensor menurun menjadi 234 ADC sehingga status sistem berubah menjadi AMAN.

Hasil tersebut menunjukkan bahwa sensor MQ-135 mampu merespons perubahan kualitas udara di dalam ruang penyimpanan dan mengirimkan data secara kontinu ke *Arduino IoT Cloud*. Nilai ADC yang diperoleh digunakan sebagai indikator relatif perubahan kondisi udara dan belum dikonversi ke satuan ppm, sehingga interpretasi hasil difokuskan pada perubahan nilai sensor sebagai dasar klasifikasi kondisi ruangan.



**Gambar 5 Grafik Historis Nilai MQ-135 pada Arduino IoT Cloud**

#### d. Pengujian *Dashboard Arduino IoT Cloud* pada Web dan Android

Pengujian ini dilakukan untuk memastikan bahwa *Dashboard Arduino IoT Cloud* mampu menampilkan seluruh data hasil pembacaan sensor secara *real-time* baik melalui web browser maupun aplikasi *Arduino Cloud Remote* pada perangkat Android.

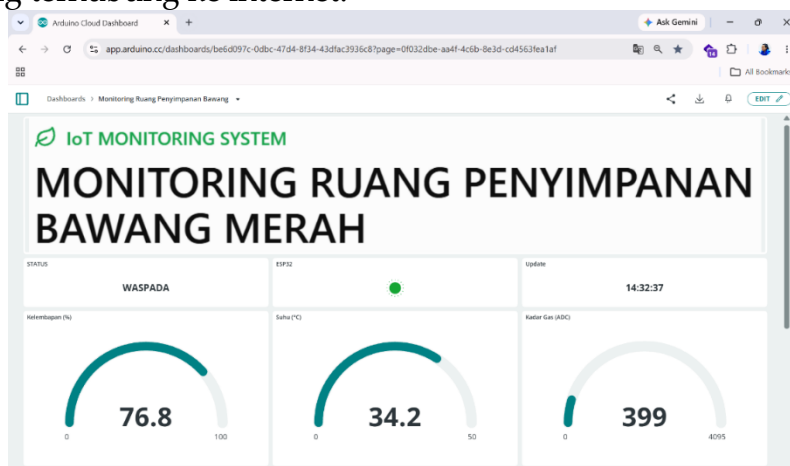
Pengujian dilakukan dengan menghubungkan *ESP32* ke *Arduino IoT Cloud*, kemudian membuka *Dashboard* menggunakan dua platform yang berbeda, yaitu web browser pada komputer dan aplikasi *Arduino Cloud Remote* pada *smartphone* Android. Selanjutnya diamati kesesuaian nilai suhu, kelembapan, kadar gas, status ruangan, status *ESP32*, dan waktu pembaruan.

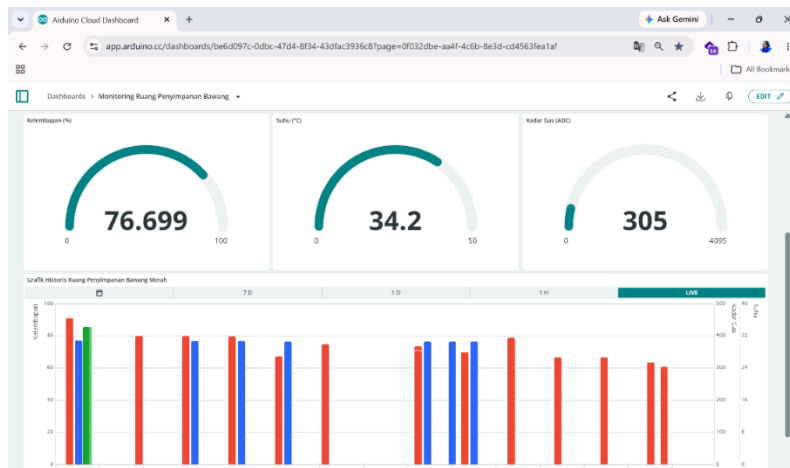
**Tabel 6 Hasil Pengujian *Dashboard* Web dan Android**

| No | Parameter Pengujian          | Web      | Android  |
|----|------------------------------|----------|----------|
| 1  | Status Ruangan               | Berhasil | Berhasil |
| 2  | Status <i>ESP32</i>          | Berhasil | Berhasil |
| 3  | <i>Update</i> Terakhir (WIB) | Berhasil | Berhasil |
| 4  | Suhu (°C)                    | Berhasil | Berhasil |
| 5  | Kelembapan (%RH)             | Berhasil | Berhasil |
| 6  | Kadar Gas (ADC)              | Berhasil | Berhasil |
| 7  | Grafik Historis              | Berhasil | Berhasil |

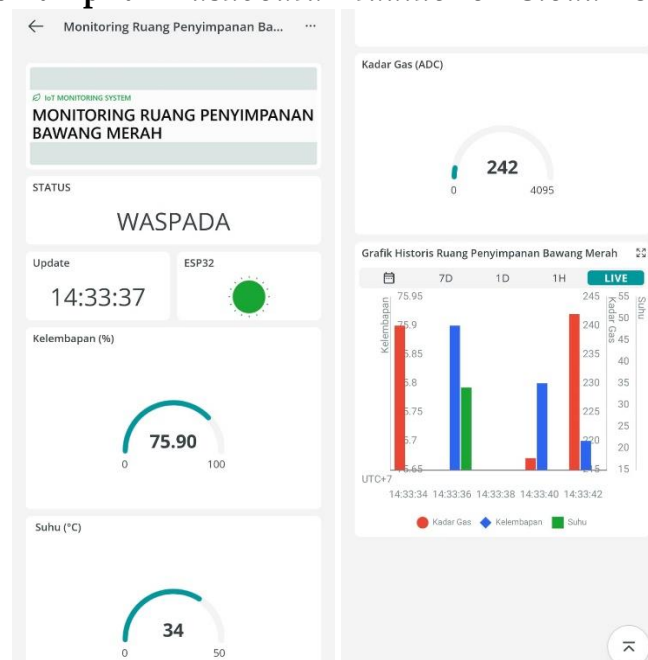
Berdasarkan Tabel 6, seluruh *widget* pada *Dashboard* berhasil ditampilkan dengan baik pada platform web maupun Android. Nilai suhu, kelembapan, dan kadar gas yang dikirim oleh *ESP32* muncul secara bersamaan pada kedua perangkat tanpa perbedaan informasi yang signifikan. Hal ini menunjukkan bahwa proses sinkronisasi data melalui *Arduino IoT Cloud* berjalan dengan baik.

Selain menampilkan data sensor, *Dashboard* juga memperlihatkan status perangkat (Online), kondisi ruangan (Aman/Waspada/Bahaya), serta waktu pembaruan dalam format WIB. Keberhasilan implementasi ini membuktikan bahwa sistem mampu melakukan pemantauan jarak jauh (*remote monitoring*) melalui berbagai perangkat yang terhubung ke internet.





**Gambar 6 Tampilan Dashboard Arduino IoT Cloud Berbasis Web**



**Gambar 7 Tampilan Dashboard Arduino IoT Cloud pada Android**

#### e. Pengujian Sistem Notifikasi

Pengujian sistem notifikasi dilakukan untuk memastikan bahwa *Arduino IoT Cloud* mampu mengirimkan *push notification* ke perangkat Android ketika kondisi ruang penyimpanan mencapai status Bahaya berdasarkan parameter yang telah ditetapkan.

Pengujian dilakukan dengan mengubah kondisi lingkungan di dalam ruang penyimpanan hingga salah satu parameter melebihi batas kategori Bahaya. Ketika variabel peringatan berubah menjadi *True*, sistem secara otomatis mengaktifkan trigger pada *Arduino IoT Cloud* dan mengirimkan notifikasi ke *smartphone* yang telah terhubung dengan akun pengguna.

**Tabel 4 7 Hasil Pengujian Sistem Notifikasi**

| No | Skenario Pengujian | Status Trigger | Hasil Notifikasi |
|----|--------------------|----------------|------------------|
|----|--------------------|----------------|------------------|

|   |                                 |              |                   |
|---|---------------------------------|--------------|-------------------|
| 1 | Kondisi ruang berstatus Aman    | <i>False</i> | Tidak terkirim    |
| 2 | Kondisi ruang berstatus Waspada | <i>False</i> | Tidak terkirim    |
| 3 | Kondisi ruang berstatus Bahaya  | <i>True</i>  | Berhasil terkirim |

Berdasarkan Tabel 7, sistem notifikasi bekerja sesuai dengan logika yang telah diimplementasikan pada *Arduino IoT Cloud*. Notifikasi hanya dikirim ketika variabel peringatan bernilai *True*, yaitu pada saat kondisi ruang penyimpanan dikategorikan sebagai Bahaya. Sebaliknya, pada kondisi Aman dan Waspada, sistem tidak mengirimkan notifikasi sehingga dapat mengurangi terjadinya peringatan yang tidak diperlukan (*false notification*).

Hasil pengujian menunjukkan bahwa integrasi antara *ESP32*, *Arduino IoT Cloud*, dan aplikasi *Arduino Cloud Remote* berhasil memberikan informasi kondisi ruang penyimpanan secara *real-time*. Dengan demikian, pengguna dapat segera mengetahui apabila lingkungan penyimpanan berada pada kondisi yang berpotensi menyebabkan penurunan kualitas bawang merah.



**Gambar 8** Tampilan *Push Notification* pada perangkat Android.

## KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, penelitian ini berhasil menghasilkan prototipe *Internet of Things* (IoT) untuk pemantauan kualitas udara dan kondisi lingkungan pada ruang penyimpanan bawang merah dengan menggunakan *ESP32 Dev Module*, sensor *DHT22*, sensor *MQ-135*, dan *Arduino IoT Cloud*. Prototipe yang dikembangkan mampu memantau suhu, kelembapan, dan kualitas udara secara *real-time* dengan interval pembacaan setiap 2 detik serta mengirimkan data secara otomatis ke *Arduino IoT Cloud* melalui jaringan Wi-Fi. Informasi hasil pemantauan dapat ditampilkan melalui *dashboard* berbasis web dan aplikasi Android yang memuat data suhu, kelembapan, kadar gas, status perangkat, status ruangan, dan waktu pembaruan, sehingga pengguna dapat melakukan pemantauan kondisi ruang penyimpanan dari jarak jauh. Selain itu, sistem notifikasi pada *Arduino IoT Cloud* mampu memberikan peringatan ketika kondisi

lingkungan mencapai kategori bahaya. Dengan demikian, prototipe yang dikembangkan dapat digunakan sebagai media pemantauan dini terhadap perubahan kualitas udara dan kondisi lingkungan pada ruang penyimpanan bawang merah.

### DAFTAR PUSTAKA

- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys and Tutorials*, 17(4), 2347–2376. <https://doi.org/10.1109/COMST.2015.2444095>
- Alyammahi, S., Alhmoudi, A., Alawadhi, M., & Alqaydi, F. (2025). *Low-Cost IoT-Based Smart Grain Monitoring System for Sustainable Storage Management*. 90. <https://doi.org/10.3390/ecsa-12-26545>
- Anitha, M., & Kumar, L. S. (2023). Development of an IoT-Enabled Air Pollution Monitoring and Air Purifier System. *Mapan - Journal of Metrology Society of India*, 38(3), 669–688. <https://doi.org/10.1007/s12647-023-00660-y>
- Aosong Electronic. (2020). DHT22 (AM2302) Digital Temperature and Humidity Sensor Datasheet. In *Aosong Electronics Co., Ltd.* <http://www.aosong.com/en/products-22.html>
- Arduino. (2026). *Arduino Cloud*. In *Arduino Documentation*. Arduino. <https://docs.Arduino.cc/Arduino-cloud>
- Baietto, M., & Wilson, A. D. (2015). Electronic-nose applications for fruit identification, ripeness and quality grading. *Sensors (Switzerland)*, 15(1), 899–931. <https://doi.org/10.3390/s150100899>
- Banзи Massimo, & Shiloh Michael. (2022). Getting Started with *Arduino*: The Open Source Electronics Prototyping Platform. In *Maker Media*. Make Community, LLC. <https://books.google.co.id/books?id=Xd3SBQAAQBAJ>
- BPS. (2024). Badan Pusat Statistik Hortikultura. In *Badan Pusat Statistik Indonesia* (Vol. 5, Number 1). Badan Pusat Statistik Republik Indonesia. <https://www.bps.go.id/id/publication/2024/06/10/790c957ba8892f9771aeebf7/statistik-hortikultura-2023.html>
- Buyya, R., & Dastjerdi, A. V. (2016). Internet of Things: Principles and Paradigms. *Internet of Things: Principles and Paradigms*, 1–354. <https://doi.org/10.1016/C2015-0-04135-1>
- Castro-Sobrino, L., Briz-Cid, N., Figueiredo-González, M., Sieiro-Sampedro, T., González-Barreiro, C., Cancho-Grande, B., Rial-Otero, R., & Simal-Gándara, J. (2019). Impact of fungicides mepanipyrim and tetraconazole on phenolic profile and colour of Mencía red wines. *Food Control*, 98, 412–423. <https://doi.org/10.1016/j.foodcont.2018.11.035>
- diagrams.net. (2024). *diagrams.net (draw.io) Documentation*. JGraph Ltd. <https://www.diagrams.net/>
- Direktorat Jenderal Hortikultura. (2022). *Laporan Kinerja Sub Sektor Hortikultura Tahun 2022*. Kementerian Pertanian Republik Indonesia.

- Espressif System. (2024). *Espressif Systems, ESP32 Technical Reference Manual*. Espressif. <https://www.espressif.com/en/support/documents/technical-documents>
- FAO. (2019). Food & Agriculture. In *Chemistry and Industry (London)* (Vol. 83, Number 2). [https://doi.org/10.1002/cind.832\\_8.x](https://doi.org/10.1002/cind.832_8.x)
- Fraden, J. (2016). Handbook of modern sensors: Physics, designs, and applications. In *Handbook of Modern Sensors: Physics, Designs, and Applications*. <https://doi.org/10.1007/978-3-319-19303-8>
- Garg, R., Kumar, A., Singh, S., & Dayana, R. (2025). Advanced Air Quality Monitoring System using IoT and Sensor Technology. *2025 3rd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, 9(4), 687–692. <https://doi.org/10.1109/IDCIOT64235.2025.10915106>
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1016/j.future.2013.01.010>
- Hanwei Electronics. (2019). *MQ-135 Gas Sensor Technical Data* (Vol. 1, pp. 3–5). Hanwei Electronics Co., Ltd. <http://www.hwsensor.com>
- Hill, W., & Horowitz, P. (2015). Art of Electronics chapter 9: Voltage regulation and power conversion. In *International Symposia in Economic Theory and Econometrics* (Vol. 16).
- Kader, A. A. (2002). *2002-Kader-Postharvest Technology of Horticultural Crops* (3rd ed.) (3rd ed.). University of California Agriculture and Natural Resources.
- Kim, J. Do, Kang, J. E., & Kim, B. S. (2020). Postharvest disease control efficacy of the polyene macrolide lucensomycin produced by *Streptomyces plumbeus* strain CA5 against gray mold on grapes. *Postharvest Biology and Technology*, 162. <https://doi.org/10.1016/j.postharvbio.2019.111115>
- Kitinoja, L., & Saran, S. (2020). Postharvest loss estimates and their impact on global food security. *Horticulturae*, 6(3), 33. <https://doi.org/10.3390/horticulturae6030033>
- Kurniawan, A. (2019). *Internet of Things Projects with ESP32*.
- Lady Ada. (2025). *DHT11, DHT22 and AM2302 Sensors*. Adafruit. <https://learn.adafruit.com/dht>
- Maier, A., Sharp, A., & Vagapov, Y. (2017). Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. *2017 Internet Technologies and Applications, ITA 2017 - Proceedings of the 7th International Conference*, 143–148. <https://doi.org/10.1109/ITECHA.2017.8101926>
- Margolis, M., Jepson, B., & Weldin, N. R. (2020). *Arduino Cookbook*. In *Online* (3rd ed.). O'Reilly Media, Incorporated.
- Miller, J. N., & Miller, J. C. (2018). *Statistics and chemometrics for analytical chemistry* (7th ed.). Pearson education.
- Minerva, R., Biru, A., & Rotondi, D. (2015). Towards a Definition of the Internet of Things (IoT). *IEEE Internet Initiative*, 1–86.

- Monk, S. (2023). Programming *Arduino*. In *New York: McGraw-Hill* (Vol. 34, Number 2). [http://fraser.stlouisfed.org/docs/historical/martin/martin55\\_1019.pdf.%0Ahttp://2.droppdf.com/files/XdTyp/programming-Arduino-getting-started-with-sketches.pdf](http://fraser.stlouisfed.org/docs/historical/martin/martin55_1019.pdf.%0Ahttp://2.droppdf.com/files/XdTyp/programming-Arduino-getting-started-with-sketches.pdf)
- Nahidul Islam, M., Wang, A., Pedersen, J. S., Sørensen, J. N., Körner, O., & Edelenbos, M. (2019). Online measurement of temperature and relative humidity as marker tools for quality changes in onion bulbs during storage. *PLoS ONE*, 14(1). <https://doi.org/10.1371/journal.pone.0210577>
- Pratama, G. A., & Nurpulaela, L. (2024). Pengaruh Suhu Pada Kinerja Sensor MQ-135 Dalam Mendeteksi Gas CO 2 Galuh Andika Pratama 1 , Lela Nurpulaela 2 Program Studi Teknik Elektro, Fakultas Teknik, Universitas Singaperbangsa Karawang. 10(20), 350–358.
- Pressman, R. S., & Maxim, B. R. (2019). *Software Engineering: A Practitioners Approach* (9th ed.). McGraw-Hill Education. <https://books.google.co.id/books?id=taIKxAEACAAJ>
- Rahman, M. S. (2020). *Handbook of food preservation* (3rd ed.). CRC Press.
- Saptadi, A. H. (2014). Perbandingan Akurasi Pengukuran Suhu dan Kelembaban Antara Sensor DHT11 dan DHT22. *JURNAL INFOTEL - Informatika Telekomunikasi Elektronika*, 6(2), 49. <https://doi.org/10.20895/infotel.v6i2.16>
- Sutton, R. S., & Barto, A. G. (2020). Reinforcement learning: An introduction (2nd ed.). In *Reinforcement learning: an introduction* (Vol. 13, Number 1). <http://incompleteideas.net/book/RLbook2020.pdf>
- Thompson, J. F., Mitchell, F. G., & Rumsay, T. R. (2008). *Commercial Cooling of Fruits, Vegetables, and Flowers*. University of California, Agriculture and Natural Resources. <https://books.google.co.id/books?id=jM3eeW5E5u0C>
- Tripathi, P. C., & Lawande, K. E. (2019). Onion storage in tropical region — a review. *Current Horticulture*, 7(2), 15. <https://doi.org/10.5958/2455-7560.2019.00014.1>
- Utami, B. K., Fajri, M. R., Johan, & Chaerani, A. D. (2025). Sistem Internet Of Things (IoT) Untuk Pemantauan Kualitas Udara Dalam Ruangan Jurnal Sains dan Teknologi. 02(02), 25–29.
- Wang, A., Casadei, F., Johansen, A., Bukman, H., & Edelenbos, M. (2016). Emission of volatile organic compounds from healthy and diseased onions. *Acta Horticulturae*, 1144, 333–339. <https://doi.org/10.17660/ActaHortic.2016.1144.49>
- Wang, A., Luca, A., & Edelenbos, M. (2019). Emission of volatile organic compounds from yellow onion (*Allium cepa* L.) bulbs during storage. *Journal of Food Science and Technology*, 56(6), 2940–2948. <https://doi.org/10.1007/s13197-019-03764-z>
- Wanyonyi, E. N. (2025). A Smart IoT-Based Real-time System for Monitoring Storage Conditions of Maize Grains. *East African Journal of Information Technology*, 8(1), 146–155. <https://doi.org/10.37284/eajit.8.1.3058>
- Wilson, A. D., & Baietto, M. (2011). Advances in electronic-nose technologies developed for biomedical applications. *Sensors*, 11(1), 1105–1176. <https://doi.org/10.3390/s110101105>