



Internet of Things-Based Automated Honey Harvesting: A Smartphone-Integrated Solution for Efficient Beekeeping

Ari Devianto^{1*}, Nur Alif Mardiyah², Muhammad Nasar³

¹Undergraduate Program in Electrical Engineering, Faculty of Engineering, Universitas Muhammadiyah Malang, Indonesia

¹PT Daesang Ingredients Indonesia, Gresik, East Java, Indonesia

^{2,3}Department of Electrical Engineering, Faculty of Engineering, Universitas Muhammadiyah Malang, Indonesia

³Department of Electrical Power Engineering, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary

aridevianto.id@gmail.com^{1*}

Abstract

Background: The low level of technological adoption in honeybee farming in Indonesia has directly impacted productivity, despite the country's vast potential as a leading global honey producer due to its rich tropical rainforests. Since 2013, Indonesia has imported approximately 70% of its national honey demand. Therefore, technological innovation in beekeeping has become an urgent necessity. **Objective:** This study aims to design and develop an automated honey harvesting system based on the Internet of Things (IoT), integrated with a smartphone application. **Research Method:** This research is an experimental design and development study that integrates traditional beehives with mechanical and electronic technologies, including a DC motor, load cell sensor to measure honey weight, DHT11 sensor to monitor environmental temperature and humidity, Arduino Uno as the main controller, and a Wi-Fi module to interface with the smartphone application. The system is powered by solar energy to support operation in remote areas. **Research Results:** The result of this design is a prototype of an automatic honey harvesting device that enables real-time monitoring and control of the harvesting process via a smartphone application. **Conclusion and Recommendations:** This innovation is expected to improve the efficiency and productivity of honeybee farming, while also promoting digital transformation within Indonesia's beekeeping sector.

Keywords: Honey, Harvest, Automatic, Internet of Things

1. Introduction

Honey is one of the leading non-timber forest products in the forestry sector. With a total land area of approximately 187.918.37 million hectares and forest areas covering 96.490.8 million hectares (51.53%), Indonesia possesses vast land resources for the development of the honey industry. The country's natural environment is highly conducive to beekeeping due to the abundance of flowering plants—at least 115 species serve as nectar sources. This favorable condition allows honey production in Indonesia to occur throughout the year [1].

From an economic perspective, honey holds significant potential as a business opportunity due to its relatively high selling price in the Indonesian market [2]. Given this potential, Indonesia should be well-positioned to become a leading honey industry nation. However, this potential has not yet been optimally realized. According to statistical data from the Directorate General of Watershed Management and Forest Protection (BPDASPS) in 2013, Indonesia's honey production was only around 2,000 tons per year, despite a large domestic market demand. With a population of approximately 250 million and an estimated per capita honey consumption of 30 grams per year, Indonesia requires at least 7,500 tons of honey annually [1].

One of the main reasons for the low production and quality of honey in Indonesia is the limited mastery of honeybee cultivation technology. The honey harvesting process is still carried out using traditional methods, involving numerous time-consuming manual steps such as opening the hive, smoking the bee colony, and extracting honey through centrifugal means. This process is estimated to consume around 90% of a beekeeper's working time solely for harvesting [3], making it inefficient and discouraging for individuals interested in beekeeping. Therefore, modern technological innovations are urgently needed to support the sustainability of honeybee farming in Indonesia [3].

The advancement of modern technology has opened new opportunities for improving the efficiency and effectiveness of honeybee farming. Appropriate technology enables beekeeping to be carried out through more practical and automated methods, replacing the previously complex and manual traditional practices. One such innovation is "Honeycomb," a student creativity program initiated in 2017 with support from the Ministry of Research, Technology, and Higher Education of the Republic of Indonesia. Honeycomb is a synthetic beehive made

from plastic materials that mimic natural hives, equipped with a microcontroller system based on Arduino Uno, which allows for automated honey harvesting via SMS [4].

Findings from this project indicated that bees were able to adapt to the artificial hive, and that automated honey harvesting could be conducted safely without dismantling the hive, thereby minimizing the risk of bee stings [5]. However, the Honeycomb system still had several limitations: the absence of automatic notifications through a smartphone application, lack of real-time temperature and humidity monitoring around the hive, and the unavailability of an integrated system for managing harvest data [4].

Building upon that research, the technological innovation in this study focuses on the development of an electronic beehive automation system based on the Internet of Things (IoT), integrated with a smartphone application [6–7]. This system will incorporate a DC motor, weight sensor (load cell), temperature and humidity sensor (DHT11), Arduino Uno as the main controller, and a Wi-Fi/GSM module for connectivity. Power will be supplied via solar panels, enabling the system to operate independently in remote beekeeping areas. This technology is expected to facilitate real-time monitoring and honey harvesting, thereby enhancing productivity and supporting the sustainable development of beekeeping in Indonesia [5, 8, 9].

2. Research Method

This study is technological engineering research employing an experimental engineering design approach [10]. This design was chosen because the primary objective of the study is to design and develop an automated honey harvesting device based on the Internet of Things (IoT), which can be integrated with a smartphone application to facilitate the harvesting process and monitoring of honey production. The research was conducted from May to June 2018.

Data Collection and Analysis

The initial stage of tool development began with data collection through literature review, market surveys, analysis of existing technologies, and field observations. The literature review involved studying books, statistical data, and online research findings to establish both theoretical and technical foundations. Market surveys were conducted to determine the availability and pricing of the required materials and components [10–11]. Further analysis focused on the "Honeycomb Synthetic Beehive Based on Arduino Uno" technology to understand its operational mechanisms and weaknesses as a basis for evaluating and improving the new system. A field study was then conducted at the "Wisata Petik Madu" honey farm in Karangploso, Malang Regency, to obtain direct data regarding the behavior and characteristics of honeybees in a real farming environment.

System Design and Tool Mechanism

The system design phase aimed at mapping and developed all technological components in an integrated manner, including both hardware and software (Fig 1.). The software component involved developing algorithms for the microcontroller and Android application, which control the automatic honey harvesting process according to the system's operational flow. The hardware design included the input power supply, controllers, input-output modules, and output mechanisms. The input section consisted of a Load Cell sensor integrated with the HX711 module, a temperature and humidity sensor (DHT11), Android system interface, a limit switch, and a reset button. The control unit utilized an Arduino Uno, which functioned as the central processor for the entire system. For the input-output communication, the system used both a GSM module and an ESP8266 Wi-Fi module. The output section employed relays as motor drivers to determine the rotational direction of the DC motor, enabling the opening and closing of the synthetic beehive.

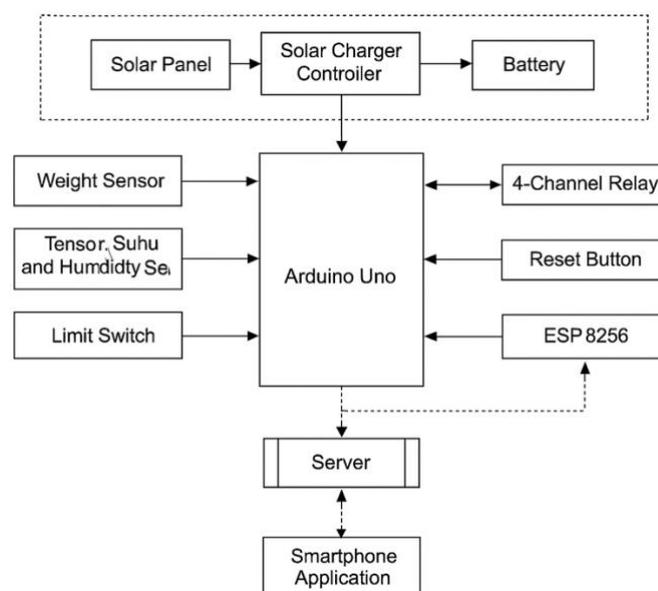


Figure 1: System Work Block Diagram

This automated honey harvesting device operates using a DHT11 sensor to monitor the temperature and humidity of the synthetic beehive, and a Load Cell sensor to measure the weight of the harvested honey. Data from both sensors are transmitted via the ESP8266 WiFi module to a server and subsequently displayed on a smartphone application. The application is equipped with a "Harvest" button, which, when pressed, sends a command to the Arduino Uno microcontroller. The Arduino then activates a DC motor connected to a gearbox and a pulley-driven lever system. This mechanism enables the synthetic hive to open and close automatically, facilitating the honey harvesting process without manual intervention. The system is powered by a solar panel connected to a rechargeable battery, managed by a smart solar charge controller to optimize energy efficiency [12–14].

Hardware and Software Design

The software design comprises three main components: the microcontroller program, the Android application, and the server system. The microcontroller programming was carried out using the Arduino IDE with C++, serving to control the DC motor that automatically opens and closes the synthetic beehive. The Android application was developed using HTML5, CSS3, and JavaScript, with the Ionic 3 framework for the front-end and ExpressJS for the back end. The server system operates on the NodeJS platform, utilizing MongoDB as the database due to its speed and ease of integration. Data exchange between the application and the server is conducted in JSON (JavaScript Object Notation) format. The hardware design includes the integration of several key components, namely the Load Cell R-NA4-2 (maximum capacity of 100 kg) with an HX711 amplifier module, DHT11 temperature and humidity sensor, ESP8266 Wi-Fi module, limit switch, reset button, and a 4-channel relay module that controls the direction of the DC motor rotation. These sensors detect the environmental conditions of the beehive and the honey weight, then send the data to the microcontroller to be processed and transmitted to the server in real time.

Testing Methodology

The testing phase aims to determine the procedures for evaluating the performance of the IoT-based honey harvesting automation tool integrated with a smartphone application. The tests include microcontroller testing, ESP8266 Wi-Fi module testing, Load Cell sensor testing, DHT11 temperature and humidity sensor testing, and Mechanical system testing.

3. Results and Discussion

Microcontroller Testing

Testing of the Arduino Uno microcontroller indicated that the device was functioning correctly. The operation of the module was marked by a green LED indicator, signalling that the board was powered on. During data uploading, the TX and RX indicator lights blinked, confirming active data communication between the device and the computer. The temperature and humidity sensor (DHT11) was tested via the serial monitor on the Arduino IDE. The results showed that the temperature and humidity data were successfully read in accordance with the environmental conditions during the test. The sensor output as monitor is shown in Figure 2.

```

Arduino 1.8.5
File Edit Sketch Tools Help

***
  delay(50);
}
Serial.print("Terhubung dg IP: ");
Serial.println(WiFi.localIP());
}

void ambilData(){
  if(toUino.available()){
    char a = toUino.read();
    if(a == ','){
      if (baacaString.length() > 1) {
        int n = baacaString.toInt();
        if(bacaString.indexOf('a') >0){
          subu = n;
          Serial.print("subu: ");
          Serial.println(subu);
        }else if (baacaString.indexOf('b') >0){
          kelembaban = n;
          Serial.print("kelembaban: ");
          Serial.println(kelembaban);
        }else if (baacaString.indexOf('c') >0){
          berat = n;
          Serial.print("berat: ");
          Serial.println(berat);
        }
      }
    }
  }
}
  
```

```

COM8 (Arduino/Genuino Uno)

Serial arduino ready
Serial to esp ready
subu: 24
Kelembaban: 68
Berat: 0
subu: 24
Kelembaban: 68
Berat: 0
subu: 25
Kelembaban: 67
Berat: 0
subu: 24
Kelembaban: 68
Berat: 0
subu: 26
Kelembaban: 68
Berat: 0
  
```

Figure 2: Test Result of Arduino Uno Microcontroller Displayed on Serial Monitor

ESP8266 Module Testing

The testing of the ESP8266 module demonstrated that the module functions optimally. It automatically connects to the designated access point (AP) network based on the configured SSID and password programmed into the system. If the module has previously been connected, the connection credentials are stored in the internal EEPROM memory, enabling it to automatically reconnect when power is restored. During the data transmission testing phase, the DHT11 sensor served as the source of temperature and humidity data. The results showed that the data displayed on the serial monitor and the data received by the server were consistent and accurate. The average time required for the data to be transmitted from the ESP8266 to the server was approximately 5 seconds. A visual representation of this testing result is shown in Figure 3

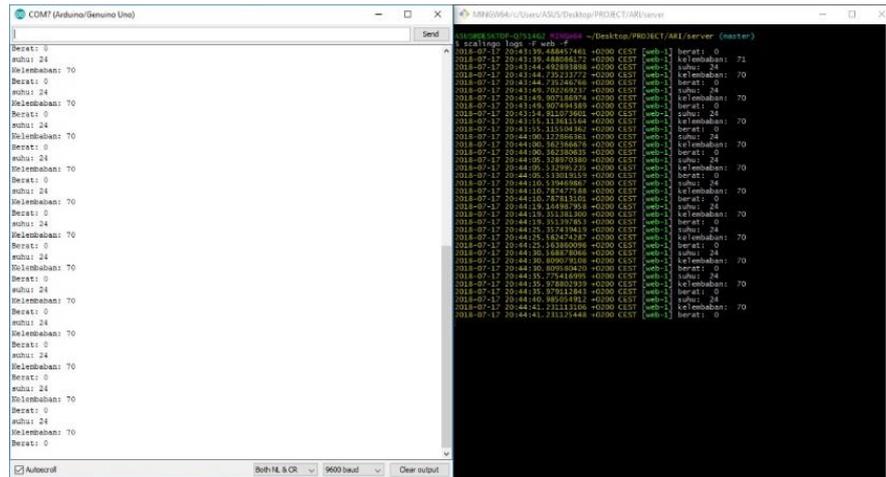


Figure 3: Results of Data Transmission Test from DHT11 Sensor via ESP8266

Load Cell and DHT11 Sensors Testing

The Load Cell sensor was tested by comparing its weight readings with those of a digital scale, used as the standard measurement tool. Two test conditions were evaluated: (1) all honeycombs empty, and (2) one honeycomb filled with honey. The results showed that the Load Cell readings were sufficiently accurate, with a measurement error of less than 5%. The observed deviation was within acceptable tolerance for an automated honey weight monitoring application. From Table 1, it can be concluded that the Load Cell sensor performance falls within a satisfactory range. The deviation between the Load Cell and digital scale measurements remains relatively small. The measurement errors likely stem from sensor reading imprecision and calibration constant mismatches. Therefore, further calibration is recommended to enhance the accuracy of weight detection and ensure proximity to actual values.

Table 1: Load Cell Sensor Test Results

| Honeycomb Condition | Digital Scale (Kg) | Load Cell (Kg) | Percentage Error |
|-------------------------------------|--------------------|----------------|------------------|
| 4 empty honeycombs | 5.1 | 5.0 | 1.96% |
| 3 empty + 1 honey-filled honeycombs | 5.8 | 6.0 | 3.44% |

DHT11 Sensor Testing

To assess the performance of the DHT11 temperature and humidity sensor, its readings were compared to those of a digital hygrometer. The results of the two test conditions (empty vs. partially honey-filled honeycombs) are displayed in Table 2. Based on Table 2, the DHT11 sensor demonstrates reasonably accurate performance, with measurement errors within the acceptable threshold of $\leq 5\%$. The discrepancies may be attributed to the DHT11's resolution, the accuracy of the reference hygrometer, and the data sampling method used by the Arduino Uno microcontroller. Overall, the DHT11 sensor is deemed reliable for environmental monitoring of synthetic beehive conditions.

Table 2: Hygrometer and DHT11 Sensor Test Results

| Honeycomb Condition | Measurement | Hygrometer | DHT11 Sensor | Error (%) |
|--------------------------------|------------------|------------|--------------|-----------|
| 4 empty honeycombs | Temperature (°C) | 26.5 | 27 | 1.88% |
| | Humidity (RH%) | 61 | 60 | 1.63% |
| 3 empty + 1 honey-filled combs | Temperature (°C) | 28.7 | 28 | 2.43% |
| | Humidity (RH%) | 61 | 58 | 4.91% |

Mechanical System Testing

The results of the mechanical system test showed that the automated honey harvesting tool functioned according to the design specifications. When the "Harvest" button on the smartphone application was pressed, the DC motor rotated counterclockwise, driving the gearbox, which in turn pulled the actuator string. This movement shifted the metal lever and opened the previously closed synthetic honeycomb cells, as illustrated in Figure 4. The DC motor used has a torque capacity of 50 kg, significantly exceeding the system's maximum load of approximately 30–35 kg when all combs are fully filled with honey. This ensures smooth operation of the system without any notable mechanical resistance. Once the cells are fully opened, the gearbox rotation triggers the limit switch, automatically stopping the motor.

Conversely, when the "Stop Harvest" button is pressed, the DC motor rotates in the opposite direction (clockwise), retracting the actuator string to return the metal lever to its initial position. The synthetic honeycomb cells close, and upon reaching the maximum closure position, the lever presses another limit switch, which halts the motor rotation. This process guarantees that the system operates safely and automatically during both opening and closing of the honeycomb cells [13].

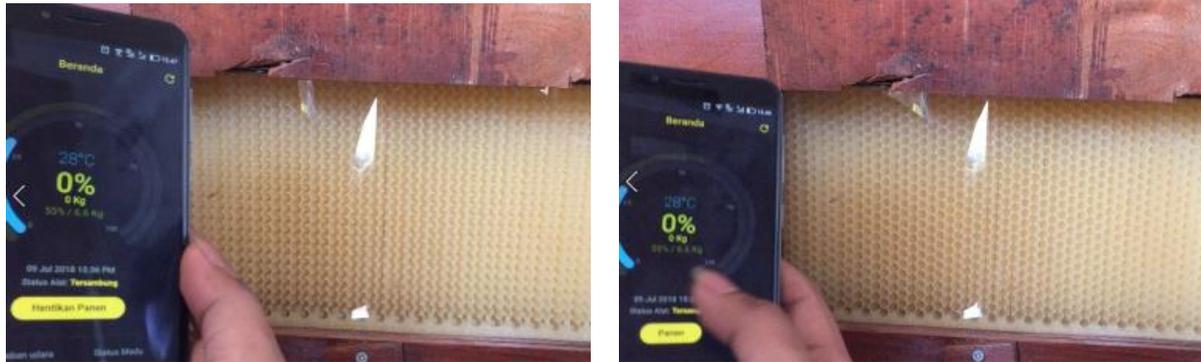


Figure 5: Mechanical System Testing

Full-System Testing of the Automated Honey Harvesting Tool

Comprehensive testing of the automated honey harvesting system demonstrated that the system operates effectively according to its intended functions. All integrated components—sensors, microcontroller, Wi-Fi module, mechanical actuators, and the smartphone application—worked in harmony throughout the harvesting cycle. When environmental conditions (temperature, humidity, and weight) reached predetermined thresholds, data from the DHT11 and Load Cell sensors were successfully transmitted via the ESP8266 module to the server and visualized through the smartphone application. The "Harvest" command triggered by the user through the app was successfully received and executed by the Arduino Uno microcontroller, initiating the DC motor's operation for honeycomb cell opening. The Load Cell sensor readings during this system-wide testing are presented in Figure 6(a).

The Load Cell sensor successfully measured the total weight of the four synthetic honeycomb cells in an empty state, yielding a value of 5 kg. To ensure that the system accurately reflects the net weight of the harvested honey, an offset calibration was applied. This calibration treated the 5 kg reading as a zero baseline within the application interface. Such adjustment allows the application to display only the actual weight of the honey, excluding the weight of the empty honeycombs. This implementation guarantees accurate and field-representative data for users.

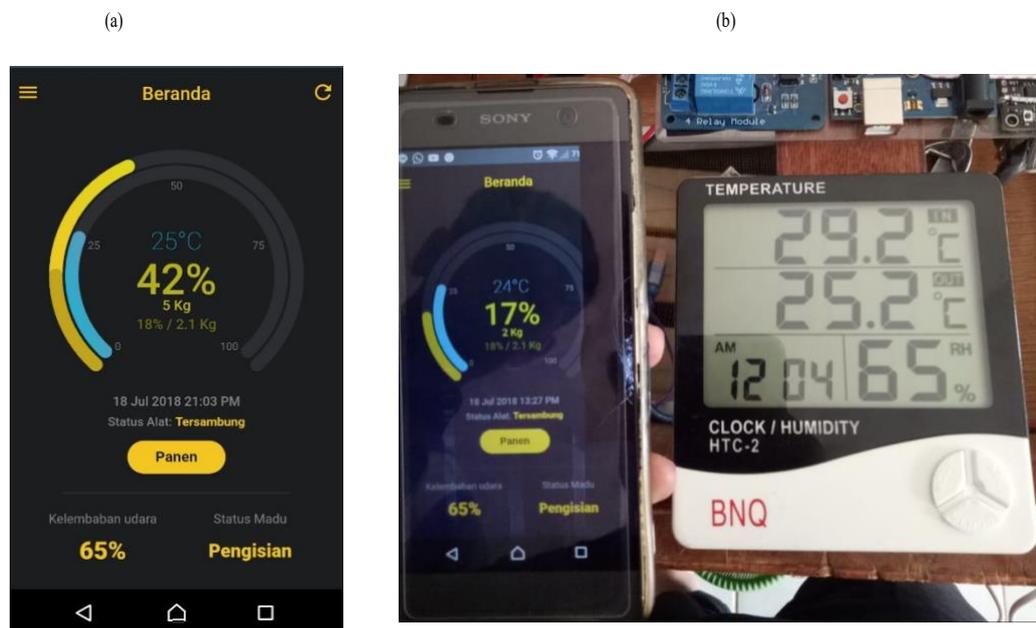


Figure 6: (a) Load Cell Sensor Readings (b) DHT11 Sensor Readings

Testing of the DHT11 sensor on empty synthetic honeycomb cells showed a temperature reading of 25°C and a relative humidity of 65%. These values were compared with a reference hygrometer, which recorded 25.2°C and 65%, respectively. The minimal discrepancies between the two sets of measurements indicate that the DHT11 sensor possesses sufficient accuracy for environmental monitoring within this system. Furthermore, results confirmed that the honey harvesting mechanism operated as designed. When the "Harvest" button on the application was pressed, the DC motor rotated counterclockwise, pulling the hook string to shift the metal lever and open the synthetic honeycomb cells. Upon reaching the maximum open position, the gear mechanism triggered a limit switch, automatically stopping the motor. Conversely, when the "Stop Harvest" button was pressed, the motor rotated clockwise, retracting the hook string and closing the honeycomb cells. The motor again stopped automatically once the closed position was reached, as indicated by the limit switch being reactivated.

4. Conclusion

Based on the testing and analysis conducted, it can be concluded that the Internet of Things (IoT)-based automatic honey harvesting system operates optimally. The system is capable of accurately reading data from the Load Cell and DHT11 sensors and transmitting the data to the server at a stable interval of every 5 seconds. The Load Cell sensor demonstrated a high level of precision, with measurement deviations under 5% when compared to standard digital scales, indicating an acceptable error margin [12]. The DHT11 sensor also performed adequately in measuring temperature and humidity, with a low error rate of less than 5% compared to a reference hygrometer, falling within the acceptable tolerance range [14–15]. The mechanical system functioned as designed: when the "Harvest" command was activated via the application, the DC motor and driving mechanism successfully opened the honeycomb cells automatically and closed them again when the "Stop Harvest" command was issued. These results confirm that both the hardware and software components of the system work synergistically and fulfil the intended design objectives [13].

Acknowledgements

The authors would like to express their sincere gratitude to all members of the research and development team who contributed to the development of this IoT-based automatic honey harvesting system. Special thanks are extended to the Department of Electrical Engineering, Faculty of Engineering, University of Muhammadiyah Malang, for their support in the form of facilities, hardware, and access to data throughout the design and testing phases. Appreciation is also directed to the beekeeping site in the Wisata Petik Madu area, Karangploso, Malang Regency, for providing valuable technical input, field experience, and for serving as a trial partner in this project.

Conflict of Interest

The authors declare no conflict of interest in this study. All phases of system design, implementation, data analysis, and manuscript preparation were conducted independently and without external influence that could compromise the integrity or interpretation of the research findings.

References

- [1] Kementerian Kehutanan, Statistik Kehutanan Indonesia Tahun 2014. Jakarta: Direktorat Jenderal Planologi Kehutanan, 2015.
- [2] S. R. Susilowati, "Peranan Hutan Dalam Menunjang Kesejahteraan Masyarakat," *Jurnal Analisis Kebijakan Kehutanan*, vol. 6, no. 3, pp. 189–204, 2009.
- [3] L. Wahyudi, A. Hadiyanto, and A. R. Hidayat, "Teknologi Panen Madu Lebah Hutan," *Jurnal Hasil Hutan*, vol. 30, no. 1, pp. 45–52, 2012.
- [4] R. Nurhuda, A. Devianto, and M. Cholid, "Honeycombic-Sarang Lebah Sintetis Menggunakan Arduino Uno Berbasis SMS Gateway, PKM-KC, 2016.
- [5] A. Devianto, S. S. Widodo, and D. Ardi, "Sistem Pemanenan Madu Otomatis Berbasis Internet of Things," *Jurnal Ilmiah Teknik*, vol. 4, no. 1, 2021.
- [6] M. I. Mahali, "Smart Door Locks Based On Internet Of Things Concept With Mobile Backend As A Service," *ELINVO*, vol. 1, no. 3, 2016.
- [7] R. Agusdian, "Penguatan Pembangunan Ekonomi Berkelanjutan melalui UMKM dan Koperasi dalam Memberdayakan Petani Madu Hutan di Sekitar Kawasan Taman Nasional Ujung Kulon," *Jurnal Sospol*, vol. 3, no. 2, pp. 122–138, 2014.
- [8] D. Putra and Mudjahidin, "Rancang Bangun Sistem Informasi Monitoring Berat dan Suhu pada Sarang Madu Berbasis Mikrokontroler," *Jurnal Teknik Industri*, vol. 11, no. 1, 2010.
- [9] G. B. Nicodemus Ujudeda, "Perancangan Teknologi Private Cloud Computing Untuk Monitoring dan Pengawasan Laboratorium Komputer di Universitas Advent Indonesia," *Jurnal TelKa*, vol. 6, no. 1, 2016.
- [10] Y. Adalina, "Analisis Finansial Usaha Lebah Madu Apis Mellifera L.," *Pusat Litbang Hutan dan Konservasi Alam*, vol. V, no. 3, pp. 217–237, 2008.
- [11] M. Mujetahid, "Teknik Pemanenan Madu Lebah Hutan Oleh Masyarakat Sekitar Hutan Di Kecamatan Mallawa Kabupaten Maros," *Jurnal Perennial*, vol. 4, no. 1, pp. 36–40, 2007.
- [12] Novandra and I. M. Widnyana, *Peluang Pasar Produk Perlebahan Indonesia*. Mataram: Balai Litbang Teknologi Hasil Hutan Bukan Kayu, 2013.
- [13] Sugiyanto, I. Okriawan, and Martinus, "Pembuatan Sistem Otomasi Dispenser Menggunakan Mikrokontroler Arduino Mega 2560," *Jurnal FEMA*, vol. 1, no. 2, 2013.
- [14] D. Prihatmoko, "Penerapan Internet Of Things (IoT) Dalam Pembelajaran Di Unisnu Jepara," *Jurnal SIMETRIS*, vol. 7, no. 2, 2016.
- [15] Alfian, M., "Implementasi Internet of Things (IoT) pada Sistem Monitoring Suhu dan Kelembaban Menggunakan NodeMCU," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 6, no. 2, pp. 275–282, 2020.