JURNAL TIMES
TECHNOLOGY INFORMATICS & COMPUTER SYSTEM

Volume XIV No 2, Desember 2025
*p*ISSN : 2337 – 3601
*e*ISSN : 2549 – 015X

Tersedia *online* di http://ejournal.stmik-time.ac.id

SIMULATION OF IOT MONITORING OF SOIL MOISTURE AND TEMPERATURE USING ARDUINO UNO ON TINKERCAD USING RULE-BASED ALGORITHMS

Muhammad Nidzom Imtiyaz[1)], Ferdian Wahyu Nabilla[2)], Moch Yasin[3)]
Program Studi Sistem Informasi
Universitas Islam Negeri Sunan Ampel Surabaya
Jl. Ahmad Yani No.117, Jemur Wonosari
email: 09040623064@uinsa.ac.id[1)], 09040623059@uinsa.ac.id[2)], moch.yasin@uinsa.ac.id[3)]

Abstract
Farmers and plant enthusiasts often face difficulties in maintaining the desired soil moisture level, as manual watering methods are usually ineffective and unstable. Too much or too little water can have a significant impact on plant growth. To resolve this issue, a study was conducted with the aim of designing and testing an automated system capable of monitoring soil moisture and temperature levels. This study adopted a Research and Development (R&D) approach, which was demonstrated in the form of a simulation in Tinkercad. The system was built using an Arduino Uno microcontroller, along with soil moisture and temperature sensors, as well as control logic based on rule-based (if-then) algorithms. Testing showed that this virtual prototype functioned as intended. This system can automatically activate the water pump when the sensor detects dry soil conditions (sensor value < 450) and stop it when the soil is moist or wet. Also, this setup effectively turns on the LED when it gets hotter than what we told it to watch out for (more than 32°C). The data we got tells us that this monitoring setup does its job well and is really good at doing things automatically based on the rules we set, which means it's a great way to solve the problems we have with watering plants by hand.
Key Words: Simulation, Arduino, Tinkercad, Automatic watering

1. Introduction

Rapid technological developments have brought about major changes in various fields, ranging from communication and education to socio-economics [1]. The agricultural sector has also experienced positive impacts from technological advances, even though the technology is considered less developed than in other sectors [2]. Until now, farmers and ornamental plant enthusiasts often use manual methods for watering. However [3], the application of automation technology can make the process more productive, effective, and efficient [4].

One of the main challenges in caring for plants is maintaining good soil moisture for the plants. Orchids, for example, require special attention in terms of water content [5]. Overwatering or underwatering can inhibit plant growth [6]. The right moisture conditions are a determining factor in the growth and survival of plants. Therefore, a system is needed that can monitor and maintain soil moisture levels regularly without the need for continuous manual intervention.

In addressing these issues, this study offers a simulation construct based on the Internet of Things (IoT) to detect soil moisture and temperature levels, with simulations conducted using Arduino Uno. This device is designed to automate the process of detecting soil moisture and watering when the soil shows signs of dryness by utilizing a rule-based algorithm, namely the if-then algorithm. The main emphasis of this research is on the simulation obtained through Tinkercad software. Arduino Uno was chosen because it is an open source microcontroller that is easy to use in programming [7]. Through this simulation, the concept and working logic of the system can be tested effectively before being implemented in hardware.

One approach in automatic plant watering systems is to use air humidity parameters as triggers, as has been done in previous studies [2]. The study utilized a DHT22 sensor to measure temperature and air humidity. However, this method has significant limitations because air humidity measurements do not directly reflect the moisture conditions of the planting medium (soil), and its accuracy is susceptible to external factors such as air circulation [8].

The main objective of this study is to design and validate an automatic soil moisture detection system through simulation. This process includes the design, program development, and then the creation of an output program for this tool. The main expectation is that the functionality and logic can be tested virtually to ensure that the proposed concept works well.

2. Literature Review

Arduino Uno

Arduino Uno is an open source microcontroller platform based on the ATMega328 chip [9]. This microcontroller is very popular due to its ease of use. It also has 14 digital inputs/outputs and 6 analog inputs. Another special feature of this device is that it has a USB bootloader, which makes it easy to upload programs to the microcontroller. The language used by this microcontroller is C. Code can be written and uploaded using software commonly referred to as Arduino IDE.

Sensor

The sensor is a key component in this device. It can detect changes in physical quantities in the surrounding environment [10]. The sensor used in this study is a soil moisture sensor. This sensor is specifically designed to detect the water content in the soil [11]. The moisture value is determined from the voltage generated when the sensor is running, thereby determining whether the soil is dry or wet.

Tinkercad

Tinkercad is a free and easy-to-use web-based circuit simulation and 3D design platform. In the context of electronics, Tinkercad allows users to design, assemble, and program virtual circuits using components such as Arduino, sensors, and actuators. The foremost benefit lies in the capacity to assess the program's rationale and the proficiency of the circuit's operation before its realization in physical components, consequently diminishing the potential for harm to elements and expediting the procedure of developing initial versions [12].

Internet of Things (IoT)

The Internet of Things (IoT) embodies a model where tangible items or equipment ("things") are outfitted with sensors, computer programs, and diverse technological elements that empower them to establish connections and share information with other equipment through the internet [13]. The basic architecture of IoT consists of devices (sensors), connectivity (internet), and data platforms. In this study, the IoT concept is applied by simulating a system in which soil moisture and temperature data can be monitored, which can later be developed to be accessible remotely.

Algoritma Rule Based

A rule-based system is an approach in artificial intelligence that uses a set of rules to solve problems [14]. These rules are usually formulated in the form of "IF [condition] THEN [action]". In this study, a rule-based algorithm was used to create automatic watering logic, for example: IF the sensor detects dry soil THEN the system will activate the virtual water pump.

3. Research Methodology

The type of research used is Research and Development (R&D). The R&D method is a research method used to produce a new product design, develop existing products, or test the effectiveness of a product [15]. In this context, the R&D method is used to produce a product in the form of a functional soil moisture detection system simulation. The research stages consist of planning, implementation, and testing, as illustrated in Figure 1

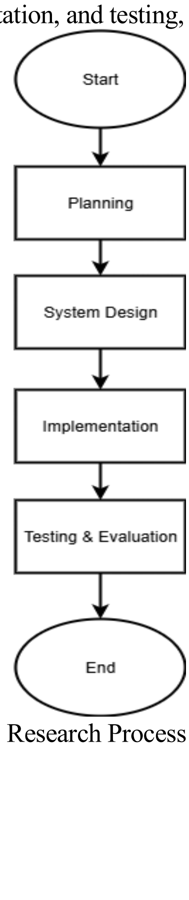

Figure 1. Research Process Flowchart

Planning

This stage begins with a needs analysis conducted through literature studies and the collection of relevant information. At this stage, the problem identified was the difficulty of manually monitoring soil moisture. Next, information was gathered on relevant components such as the Arduino Uno microcontroller, soil moisture sensor, and software to be used, namely Tinkercad as a simulation platform and Arduino IDE for programming.

System Design

After analyzing the requirements, the next step is to design a system that serves as a blueprint prior to implementation. Software design focuses on workflow and program logic, which are visualized in the form of a flowchart as shown in Figure 2. This flowchart illustrates the process from reading sensor input, evaluating conditions based on predefined rules, to producing output in the form of actions on the actuator.

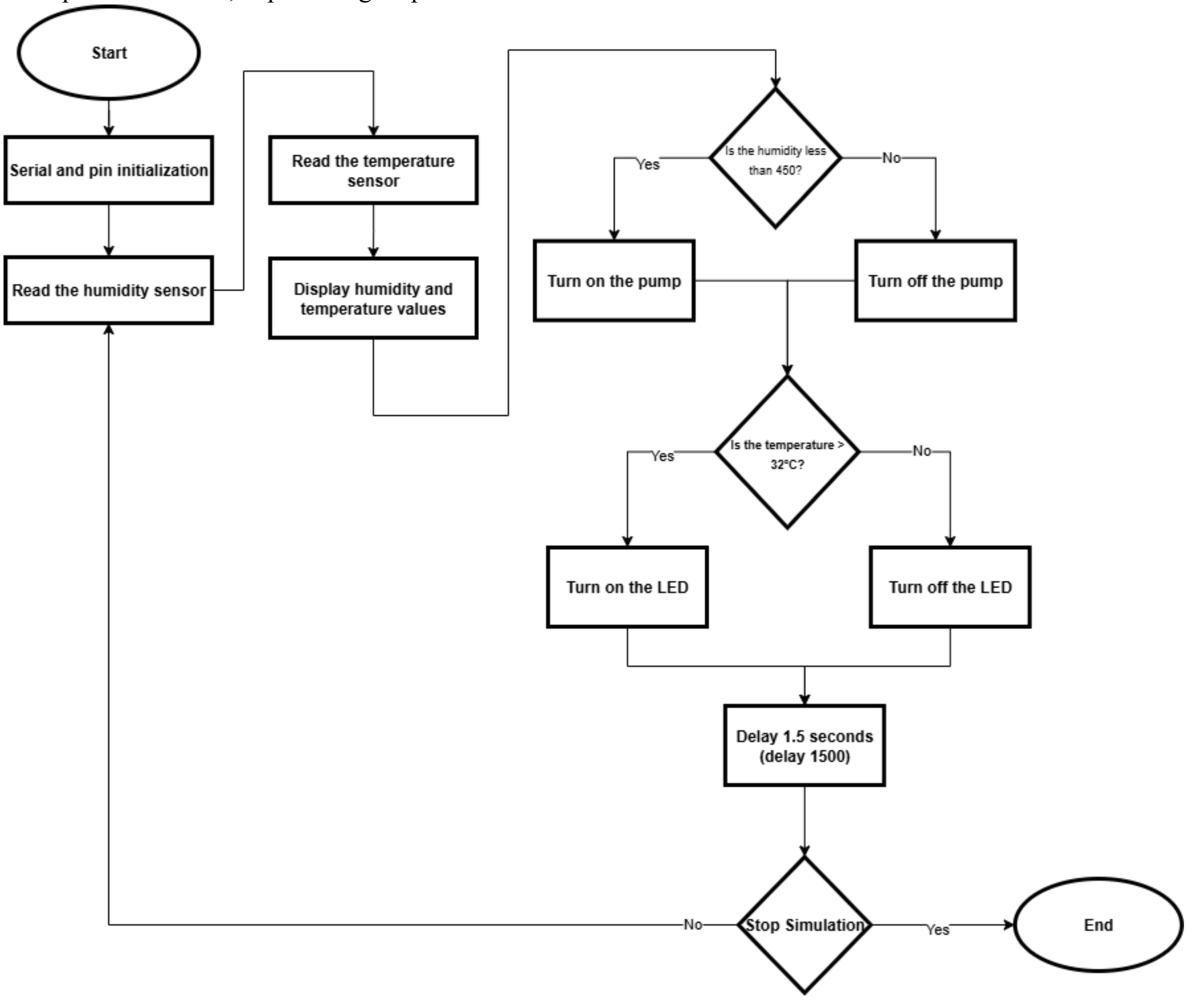

Figure 2. Flowchart of rule-based pump and LED control based on humidity and temperature sensor data

Implementation

During the implementation phase, the entire system design—both hardware and software—was realized as a functional virtual prototype within the Tinkercad platform. This phase was divided into two main activities: virtual circuit assembly and algorithm implementation on the microcontroller.

1. Virtual Circuit Assembly
   The first activity is to assemble all virtual components on a breadboard according to the circuit diagram that was established during the design stage. This involves creating virtual cable connections between the Arduino Uno microcontroller pins and each external component.
2. Algorithm Implementation and Coding
   The second activity is to embed logic into the system. The logic and workflow designed in the flowchart are translated into the Arduino programming language (C++-based) using the code editor integrated in Tinkercad. This is where the rule-based algorithm is implemented. Technically, the rules in if-else format are written in the void loop() function in the program code. After the code is uploaded, these rules are stored and executed continuously in the Arduino Uno virtual microcontroller.

Testing

The final stage is testing, where the simulation product that has been created is tested. The purpose of this testing is to assess whether the virtual product works well in accordance with the objectives that have been set. Testing is carried out by running the simulation on Tinkercad and observing the results.

4. Result

This section presents the results of the implementation and testing of the Arduino Uno-based soil moisture and temperature monitoring system simulation on the Tinkercad platform. The research results cover two main aspects: the overall functionality of the system and the application of rule-based algorithms in controlling actuators.

System Simulation Series

The hardware implementation was carried out by assembling all components—Arduino Uno, Capacitive Soil Moisture Sensor, TMP36 Temperature Sensor, 16x2 I2C LCD, DC Water Pump, and LED—on the Tinkercad simulation platform.

In this circuit, Arduino Uno receives data from two main sensors that function as inputs. The Capacitive Soil Moisture Sensor, which measures the water content in the soil, sends its analog signal to the Arduino's A0 pin. Meanwhile, the ambient temperature data is detected by the TMP36 Temperature Sensor, whose output is then directed to the Arduino's A1 pin. These two analog pins serve as gateways for the Arduino to read and process real-time environmental data.

The addition of an I2C LCD serves as an interface for displaying sensor data and system status, providing richer information than LED indicators alone. The final circuit that was Successfully implemented and became the main product of this research is presented in Figure 3.

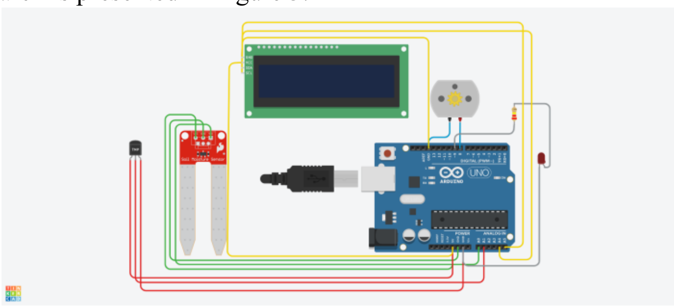

Figure 3. Simulation Circuit

Implementation of Rule-Based Logic

The rule-based algorithm that forms the core of this system is implemented directly into the Arduino program code. This logic functions as the "brain" that determines decisions based on input data from sensors. Figure 4 shows a snippet of crucial code embedded in the void loop() function that demonstrates the implementation of rules for the water pump and LED.

```
void loop() {
  // 1. READ DATA FROM SENSORS
  int moistureValue = analogRead(moistureSensorPin);

  int rawTempValue = analogRead(tempSensorPin);
  float voltage = (rawTempValue / 1024.0) * 5.0;
  float tempCelsius = (voltage - 0.5) * 100.0;

  lcd.clear();

  lcd.setCursor(0, 0);
  lcd.print("Humidity: ");
  lcd.print(moistureValue);

  lcd.setCursor(0, 1);
  lcd.print("Temp: ");
  lcd.print(tempCelsius);
  lcd.print(" C");

  // 2. RULE-BASED ALGORITHM
  if (moistureValue < dryThreshold) {
    digitalWrite(pumpPin, HIGH);
  } else {
    digitalWrite(pumpPin, LOW);
  }

  if (tempCelsius  > hotThreshold) {
    digitalWrite(tempIndicatorPin, HIGH);
  } else {
    digitalWrite(tempIndicatorPin, LOW);
  }

 delay(1500);
}
```

Figure 4. Rule-based Code

Test Results

Functional testing was conducted by providing varying input values to the soil moisture sensor and observing the system response, both on the actuator (water pump) and on the LCD screen. The testing proved that the system can operate autonomously according to the programmed rules. A summary of the test results is presented in Table 1 and Table 2.

Table 1. *Humidity System Testing*

Scenario	Humidity sensor input	Pump Action	LCD Screen Display	Status
Very Dry	100	On	Humidity Temperature	Successful
Dry Soil	300	On	Humidity Temperature	Successful
Dry Limit	400	On	Humidity Temperature	Successful
Moist Soil	500	Off	Humidity Temperature	Successful
Wet Soil	700	Off	Humidity Temperature	Successful
Very Wet Soil	800	Off	Humidity Temperature	Successful

The results in Table 1 show that the humidity control system works as designed. The water pump activates when the sensor value is below the set threshold (i.e., 450). When the sensor value reaches 500 (humid conditions), the system Successfully turns off the pump automatically. This validates that the implemented if-else logic is capable of responding to soil conditions with precision.

Table 2. *Temperature System Testing*

Scenario	Temperature sensor input (^{o})	LED Action	LCD Screen Display	Status
Cold	10	Off	Humidity Temperature	Successful
Cool	20	Off	Humidity Temperature	Successful
Normal	30	On	Humidity Temperature	Successful
Warm	40	On	Humidity Temperature	Successful
Hot	50	On	Humidity Temperature	Successful

Temperature system testing in Table 2 also demonstrated the expected functionality. The indicator LED Successfully activated when the temperature reached the normal threshold (30°C) and remained lit at higher temperatures.

Visual Verification of System Output

To provide stronger functional evidence, the system output conditions are documented visually. Figure 5 shows the simulation conditions when the temperature is below the threshold (e.g., 20°C), where the indicator LED is off. Conversely, Figure 6 shows the system response when the temperature exceeds the threshold (e.g., 40°C), where the indicator LED Successfully turns on. This visual documentation confirms that the actuator output works according to the programmed logic.

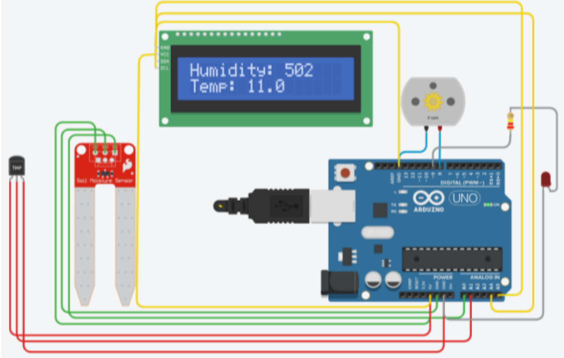

Figure 5. Screenshot of Simulation when LED is Off

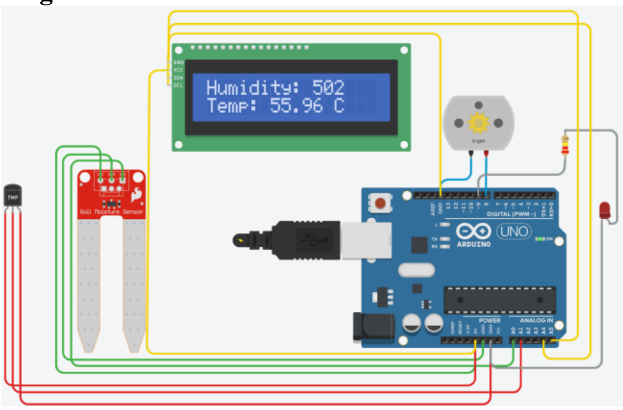

Figure 6. Screenshot of Simulation when LED is On

5. Conclusion

This research has Successfully designed and implemented a functional prototype of a virtual soil moisture and temperature monitoring system on the Tinkercad platform. The implementation of a rule-based algorithm has proven to be an effective core control system, enabling the device to work autonomously in response to environmental conditions. This success was validated through a series of tests, in which the system was consistently and accurately able to activate the water pump when the soil was detected to be dry (sensor value < 450) and turn on the indicator LED when the temperature exceeded the threshold (> 32°C). Thus, this simulation product not only proves the technical feasibility of the proposed design, but also Successfully achieves the research objective as a potential solution to overcome manual monitoring and irrigation automation problems.

To improve the system's capabilities and effectiveness going forward, several suggestions are worth considering. The first step involves converting this digital representation into real, physical components for testing in a real-world environment. Next, the system can be developed by adding connectivity modules such as ESP8266 to enable remote monitoring (IoT) features via a web dashboard or mobile application. In addition, the control logic can be refined using more adaptive algorithms such as Fuzzy Logic for higher precision, as well as the addition of data logging features for historical analysis of soil conditions over the long term.

6. Reference

[1] E. Tenda, Eric Alfonsius, Megastin M. Lumembang, and Eliasta Ketaren, "EARLY WARNING SYSTEM UNTUK POTENSI BENCANA LONGSOR DIKOTA MANADO BERBASIS INTERNET OF THINGS," J. TIMES, vol. 12, no. 2, pp. 64–70, Dec. 2023, doi: 10.51351/jtm.12.2.2023710.

[2] A. Roihan, A. Mardiansyah, A. Pratama, and A. A. Pangestu, "Simulasi Pendeteksi Kelembaban Pada Tanah Menggunakan Sensor Dht22 Dengan Proteus," Method. J. Tek. Inform. dan Sist. Inf., vol. 7, no. 1, pp. 25–30, 2021, doi: 10.46880/mtk.v7i1.260.

[3] S. Sufaidah, E. Lilawati, F. Arya Dinta, K. Khofifah, E. B. Wijanarko, and D. Cahyaningtyas, "Sosialisasi Program Penyiraman Tanaman Hias Secara Otomatis Berbasis Arduino," Jumat Inform. J. Pengabdi. Masy., vol. 5, no. 1, pp. 36–44, Jun. 2024, doi: 10.32764/abdimasif.v5i1.4243.

[4] A. L. Kalua, J. Ramadhan Kaplale, N. Klarissa Ribka Mapalie, I. Syagitsyah Iriansu, and E. Ketaren, "PENGEMBANGAN ALAT TES KESAMAPTAAN TNI/POLRI BERBASIS TEKNOLOGI COMPUTER VISION DAN IOT DENGAN METODE POSE ESTIMATION MENGGUNAKAN PYTHON," J. TIMES, vol. 13, no. 2, pp. 294–299, Dec. 2024, doi: 10.51351/jtm.13.2.2024791.

[5] F. Nuni Dea Rustanti and R. Zainul Abidin, "SISTEM MONITORING KADAR KETERSEDIAAN AIR PADA TANAMAN ANGGREK MENGGUNAKAN SENSOR SOIL MOISTURE BERBASIS INTERNET

OF THINGS (IoT),” JATI (Jurnal Mhs. Tek. Inform., vol. 8, no. 5, pp. 10197–10203, Sep. 2024, doi: 10.36040/jati.v8i5.11012.

[6] E. Nurcahyani and A. P. Sabatini, “Respon Planlet Anggrek Cattleya sp. Hasil Seleksi In Vitro Terhadap Cekaman Kekeringan dengan Polietilenglikol (PEG) 6000,” Indones. J. Biotechnol. Biodivers., vol. 6, no. 2, pp. 61–67, 2022.

[7] S. Sutono and F. Al Anwar, “Perancangan dan Implementasi Smartlamp berbasis Arduino Uno dengan menggunakan Smartphone Android,” Media J. Inform., vol. 11, no. 2, p. 36, 2020, doi: 10.35194/mji.v11i2.1036.

[8] A. P. Nanda, J. Jeprianto, and M. I. Mahdi, “SISTEM OTOMATIS PENYIRAMAN TANAMAN BERBASIS SENSOR KELEMBAPAN TANAH UNTUK PENINGKATAN PRODUKTIVITAS PERTANIAN,” Technol. J. Ilm., vol. 15, no. 4, p. 764, Oct. 2024, doi: 10.31602/tji.v15i4.16300.

[9] E. M. Punuh, “Rancang Bangun Sensor Parkir Kendaraan Roda Empat Berbasis Mikrokontroler Arduino Uno,” Jambura J. Electr. Electron. Eng., vol. 6, no. 1, pp. 18–24, Jan. 2024, doi: 10.37905/jjeee.v6i1.22171.

[10] R. Muttaqin, W. S. W. Prayitno, N. E. Setyaningsih, and U. Nurbaiti, “Rancang Bangun Sistem Pemantauan Kualitas Udara Berbasis Iot (Internet Of Things) dengan Sensor DHT11 dan Sensor MQ135,” J. Pengelolaan Lab. Pendidik., vol. 6, no. 2, pp. 102–115, May 2024, doi: 10.14710/jplp.6.2.102-115.

[11] H. Marcos and H. Muzaki, “Monitoring Suhu Udara Dan Kelembaban Tanah Pada Budidaya Tanaman Pepaya,” J. Teknol. dan Sist. Tertanam, vol. 3, no. 2, 2022, doi: 10.33365/jtst.v3i2.2200.

[12] B. N. Mohapatra, R. K. Mohapatra, J. Joshi, and S. Zagade, “Easy Performance Based Learning Of ArduinoAnd Sensors through Tinkercad,” Int. J. Open Inf. Technol., vol. 8, no. 10, pp. 73–76, 2020, [Online]. Available: http://injoit.org/index.php/j1/article/view/977

[13] S. Kumar, P. Tiwari, and M. Zymbler, “Internet of Things is a revolutionary approach for future technology enhancement: a review,” J. Big Data, vol. 6, no. 1, p. 111, Dec. 2019, doi: 10.1186/s40537-019-0268-2.

[14] J. van der Waa, E. Nieuwburg, A. Cremers, and M. Neerincx, “Evaluating XAI: A comparison of rule-based and example-based explanations,” Artif. Intell., vol. 291, p. 103404, Feb. 2021, doi: 10.1016/j.artint.2020.103404.

[15] B. Muqdamien, U. Umayah, J. Juhri, and D. P. Raraswaty, “TAHAP DEFINISI DALAM FOUR-D MODEL PADA PENELITIAN RESEARCH & DEVELOPMENT (R&D) ALAT PERAGA EDUKASI ULAR TANGGA UNTUK MENINGKATKAN PENGETAHUAN SAINS DAN MATEMATIKA ANAK USIA 5-6 TAHUN,” Intersections, vol. 6, no. 1, pp. 23–33, Feb. 2021, doi: 10.47200/intersections.v6i1.589.