



# Automatic Temperature, Lighting, and Fish Feeder Control and Monitoring Systems in Aquascape based on the Internet of Things

Raihan Eka Saputra<sup>1\*</sup>, Adi Pandu Wirawan<sup>1</sup>, Didit Suprihanto<sup>1</sup>,  
Aji Ery Burhandenny<sup>1</sup> , and Happy Nugroho<sup>1</sup>

<sup>1</sup> Electrical Engineering Department, Mulawarman University, Jalan Sambaliung No. 9,  
Samarinda, East Kalimantan, Indonesia  
raihanekasaputraa@gmail.com

**Abstract.** Aquascaping is the art of designing an aquarium that combines water, plants, fish, and decorations to create aesthetic harmony. However, maintaining an aquascape poses several challenges, particularly regarding environmental factors such as unstable water temperature and suboptimal lighting. The ideal water temperature for an aquascape ranges between 25°C and 28°C, while appropriate lighting is crucial to support photosynthesis in aquatic plants. By designing a system that integrates Internet of Things (IoT) technology, users can remotely monitor the conditions of their aquascape. This system provides a solution to maintaining the ideal environment for plants and fish in an aquascape aquarium. The system enables temperature monitoring via the Thingier.io website, with automated temperature control that adjusts the water temperature if it exceeds or falls below the ideal range. Automated lighting control adjusts the light duration according to the plants' needs, optimizes the photosynthesis process, and an automatic feeder provides fish feeding. The system is developed using NodeMCU ESP8266 for WiFi connectivity, a DS18B20 temperature sensor for temperature measurement, an RTC for timekeeping, and an LCD and website for displaying information. The performance test of the DS18B20 temperature sensor showed a Mean Absolute Percent Error (MAPE) of 1.02%. Relay testing indicated that the fan relay activates at temperatures  $\geq 28^\circ\text{C}$ , the heater relay activates at temperatures  $< 25^\circ\text{C}$  to maintain the ideal temperature, and the light relay operates from 07:00 WITA to 12:59 WITA and 16:00 WITA to 21:59 WITA to ensure optimal lighting conditions. The automatic feeding test showed that the fish consumed 0.5 grams of food in 2 minutes and 52 seconds.

**Keywords:** Aquascape, Automatic Feeder, Internet of Things (IoT), Lighting and Water Temperature

## 1 Introduction

Aquascaping, or garden aquariums, is a type of aquarium that emphasizes the combination of water, plants, fish, and other decorative elements to create harmonious aesthetics. Aquascaping is a relatively new art form that has begun to develop in Indonesia. This art form is not much different from the art of arranging aquariums, and the process of maintaining an aquascape is referred to as aquascaping [1]. The main goal of aquascaping is to create an underwater landscape within the aquarium, making it more beautiful and appealing to enhance the aesthetics of a room while also considering the maintenance of aquatic plants. However, maintaining an aquascape presents several challenges, especially regarding the stability of the environment, which often disrupts the photosynthesis process of aquatic plants. Extreme and unpredictable weather can cause the water temperature inside the tank (aquarium) to rise during the day, where the ideal water temperature for aquascaping is between 25°C and 28°C [2].

Temperature is a crucial factor that influences the life of biota in aquascapes, including fish. The optimal temperature range for fish, particularly guppies, is between 24°C and 28°C. If the water temperature falls too low or rises too high, guppies can become stressed, which negatively affects their health [3]. The feeding frequency for ornamental guppies should also be carefully monitored to avoid overfeeding, with the best practice being to feed them twice a day, in the morning and afternoon [4]. Various plant species commonly used in aquascapes, such as *Hygrophila polysperma* 'Sunset', require an ideal temperature range between 20°C and 28°C to thrive [5].

Lighting is one of the most important elements in aquascaping because it directly affects the health and growth of living organisms in the aquascape. Proper lighting is crucial as plants require light for photosynthesis and to grow healthily. Adequate lighting is essential for the growth of aquatic plants and the well-being of aquatic animals [6]. For optimal results, plants should receive light for a maximum of 12 hours per day and a minimum of 5 hours per day. Typically, the light should be on for 6 hours, off for 3 hours, and then on again for another 6 hours. Plants deprived of light will appear stunted and unhealthy. Conversely, excessive light exposure can cause algae growth, leading to dull leaves. With the right type of light and appropriate duration, plants will grow normally and healthily. Light sources such as fluorescent lights, LED lights, or metal halide lights can be used [5].

An observation conducted with an aquascape ornamental fish aquarium owner in Jl. Lumba-Lumba No.3, Kec. Samarinda Ilir, Kota Samarinda revealed that the water temperature in the aquarium is manually checked using a thermometer placed inside the tank. The challenge for the owner is that they cannot monitor the ideal water temperature when they are away from the aquarium, making it difficult to take immediate action if there is an extreme temperature fluctuation, as they are unable to access real-time temperature data.

Based on the various challenges faced by aquascape aquarium owners, the development of appropriate technology is necessary. In addition to temperature control, this research will introduce an automatic lighting system to support the photosynthesis process of aquascape plants, as well as an automatic fish feeder. The technology required must allow for real-time temperature monitoring from anywhere and will be built using

Internet of Things (IoT) technology, enabling the system to connect with the user's mobile device. The automation system will manage lighting to maintain the ideal conditions for photosynthesis, schedule fish feeding automatically, and regulate water temperature to maintain optimal conditions in the aquascape aquarium. Therefore, this research will focus on designing and developing a monitoring system and automation control for temperature, lighting, and fish feeding in aquascapes based on the Internet of Things.

## 2 Materials and Methods

### 2.1 Research Methodology

Research methodology refers to the steps and procedures that will be undertaken in data or information collection to solve problems and test research hypotheses. The method used by the author in this research is the experimental method, designed by developing a Monitoring System and Automation Control for Temperature, Lighting, and Feeder in an Aquascape Based on the Internet of Things.

### 2.2 Tools and Materials

The tools and materials used in the research for the Development of a Monitoring System and Automation Control for Temperature, Lighting, and Feeder in an Aquascape Based on the Internet of Things are listed below. The details of the required tools and materials can be seen in Table 1.

**Table 1.** Tools and Materials Required

No.	Tools/Materials	Function	Quantity
1.	Fan	Aquarium water cooling	1 Unit
2.	ESP8266	Microcontroller	1 Unit
3.	Male and Female Cables	Component connectors	1 Set
4.	4-Channel Relay	On/off condition controller	1 Unit
5.	DS18B20 Waterproof Temperature Sensor	Temperature reader sensor	1 Unit
6.	Heater	Aquarium water heater	1 Unit
7.	16x2 LCD	Displays measurement results	1 Unit
8.	Adapter	Power source for microcontroller	1 Unit
9.	Plastic Box	Component holder	1 Unit
10.	LED Light	Lighting source	1 Unit
11.	Real Time Clock (RTC)	Time module	1 Unit
12.	Servo Motor	Automatic feeder valve actuator	1 Unit

## 2.3 Research Design

The following is a detailed explanation of the research design applied:

**System Design and Development.** The system design consists of two parts hardware design and Thingier.io integration. This stage requires tools and materials such as the ESP8266 microcontroller, DS18B20 temperature sensor, heater, fan, 16x2 LCD, servo motor, and RTC. The system is powered by a 12V adapter connected to the ESP8266 microcontroller. The ESP8266 is linked to the DS3231 RTC, allowing it to read real-time data. The DS18B20 temperature sensor is also connected to the ESP8266 to monitor temperature. The 16x2 LCD displays both temperature and time. The servo motor functions to open and close the fish feeder valve located in the fish feed container. A relay connected to the heater and fan will automatically turn on and off based on the pre-set temperature conditions, while the light will turn on and off automatically according to the set schedule. The ESP8266 microcontroller connects to the internet via Wi-Fi. Once connected, Thingier.io will display temperature readings and the status of the system components on a mobile device, allowing users to remotely monitor the system. The block diagram of the system design is shown in Fig. 1 below.

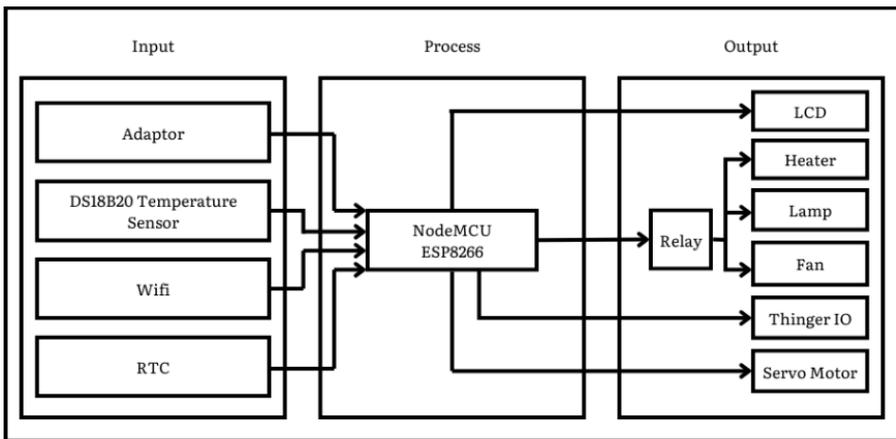
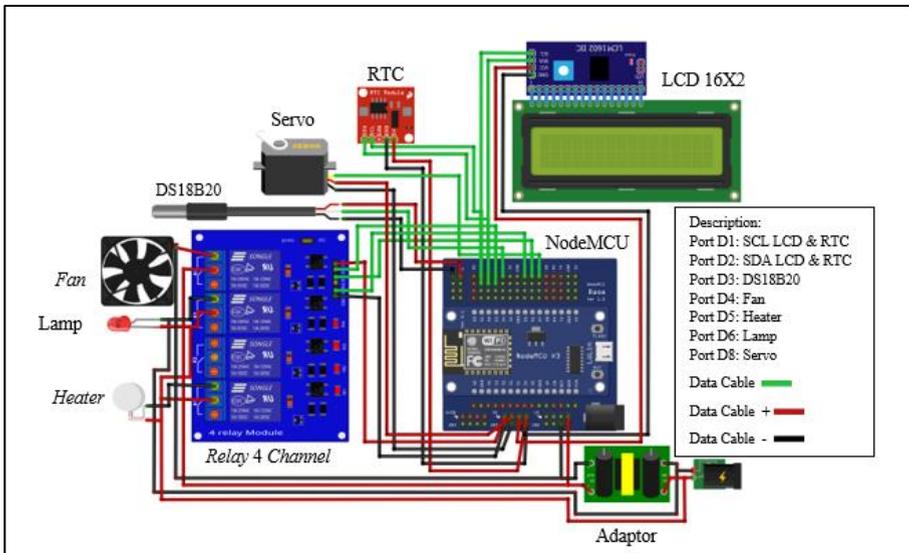


Fig. 1. Block Diagram of the System Design

**Hardware Design.** The temperature monitoring system, automatic feeder, and lighting control for the aquascape based on the Internet of Things (IoT) is built using the ESP8266 microcontroller module, DS18B20 temperature sensor, heater, fan, 16x2 LCD, servo motor, and RTC. The hardware system is designed to measure the water temperature in the aquascape aquarium. The operation of this hardware system is as follows: when the water temperature is  $\geq 28^{\circ}\text{C}$ , the ESP8266 microcontroller will signal to activate the relay connected to the fan. If the water temperature is between  $25^{\circ}\text{C}$  and  $<28^{\circ}\text{C}$ , the ESP8266 will signal to deactivate both the fan and the heater. When the water temperature is  $<25^{\circ}\text{C}$ , the microcontroller will signal to activate the relay con-

nected to the heater. Temperature changes will be continuously monitored and displayed on the 16x2 LCD as well as on mobile devices connected via Thingier.io. The lighting control system is set to automatically turn on from 07:00 WITA to 12:59 WITA, then turn off from 13:00 to 15:59 WITA. The light will then turn on again from 16:00 to 21:59 WITA and turn off from 22:00 WITA until the next day, turning back on at 07:00 WITA. This schedule ensures that the aquascape receives adequate lighting for 12 hours each day. The servo motor will automatically open to feed the fish according to the schedule in the morning at 08:00 WITA and in the evening at 17:00 WITA. This timing will follow the readings from RTC.

**Schematic Diagram.** In the overall system, the schematic diagram shows how the components are interconnected. Fig. 2 illustrates the schematic configuration of the system.



**Fig. 2.** Overall System Schematic

**Device Design.** The design of the system can be seen in Fig 3, which shows the overall appearance. The system includes an aquarium with dimensions of 30 x 15 x 25 cm, equipped with control devices, a fan, a heater, aquarium lighting, and an automatic feeding tube.

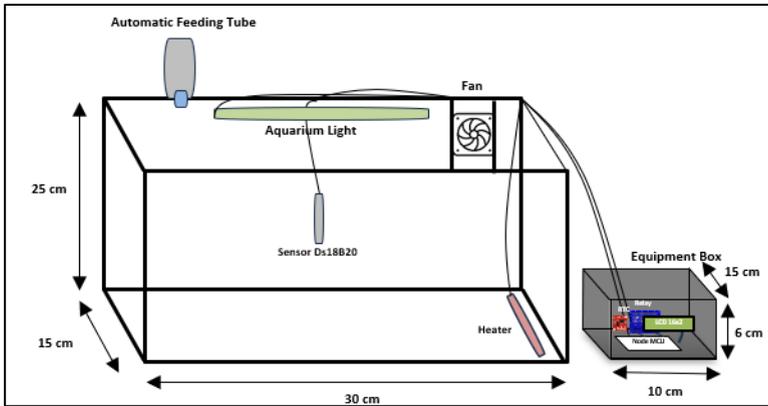


Fig. 3. Overall Device Design

The following is a flowchart of the operation of the DS18B20 temperature sensor module:

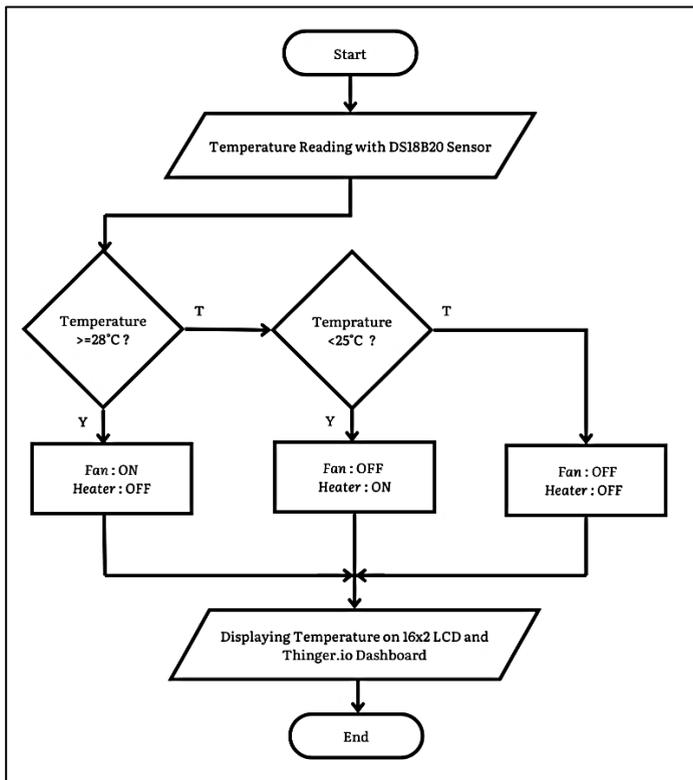


Fig. 4. Flowchart of DS18B20 Sensor Operation

The following is a flowchart of the operation of the DS3231 RTC:

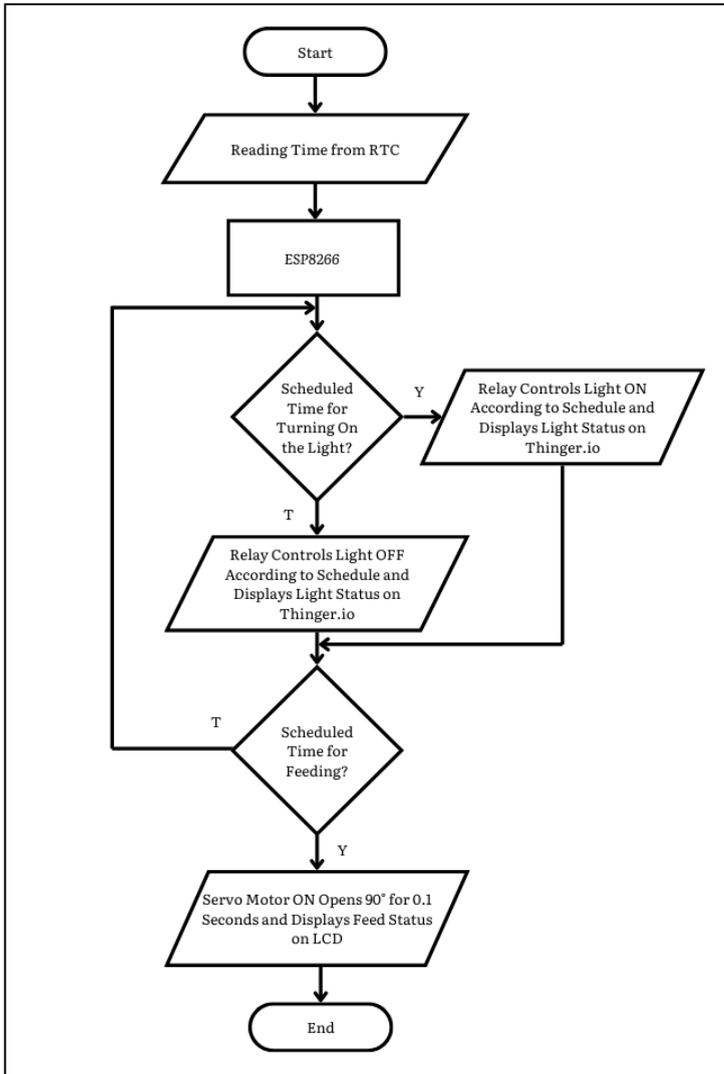


Fig. 5. Flowchart of DS3231 RTC Operation

**Thingier.io Monitoring Dashboard.** The Thingier.io monitoring dashboard functions as a platform to display real-time graphs or numerical data for water temperature conditions and relay statuses. The dashboard includes a DS18B20 temperature monitoring display, a DS18B20 temperature chart, a clock, and the status of relays for the heater, light, servo, and fan. This setup makes it easy to monitor the temperature conditions in the aquascape and the status of the various relays in the system.

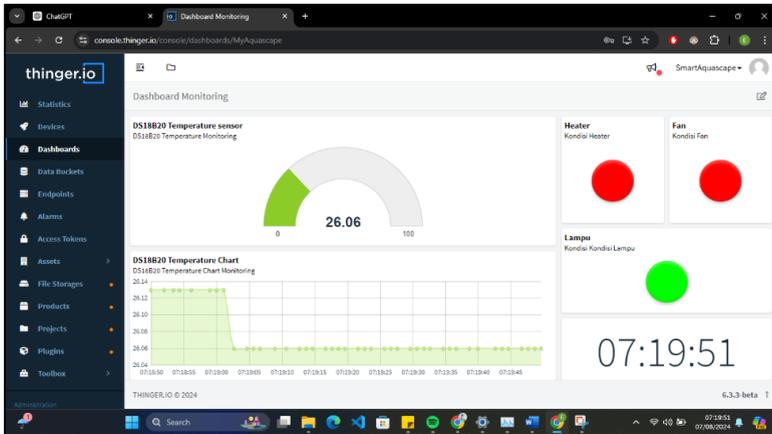


Fig. 6. Monitoring Display on Thingier.io Website

### 3 Results and Discussion

#### 3.1 Design Results

The system for monitoring temperature, lighting automation, and feeder on the aquascape based on the Internet of Things (IoT) is built from several components. It consists of a box-shaped device with dimensions of 15 cm in length, 6 cm in height, and 10 cm in width. The device is equipped with a temperature control system that includes a fan to lower the water temperature when it rises and a heater to increase the water temperature when it falls below the setpoint. A DS18B20 temperature sensor is used to monitor the water temperature in the aquascape. Additionally, an aquarium light provides illumination for the aquascape, and an automatic feeding tube is used to feed the fish automatically according to a pre-set schedule. The design results of the device can be seen in Fig. 7.

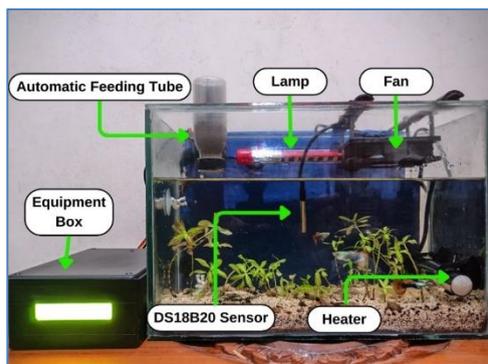


Fig. 7. Device Design Results

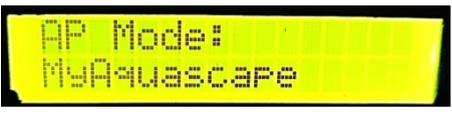
### 3.2 System Testing and Implementation

System testing is conducted to ensure that each component is ready when the system is operational. The components tested include the NodeMCU's Wi-Fi connection, DS18B20 temperature sensor, relay testing, servo motor testing, the Thingier.io dashboard, 16x2 LCD testing, and RTC DS3231 testing.

### 3.3 NodeMCU Connection Test Results on Wi-Fi

This test aims to ensure that the NodeMCU being used is in good condition and capable of connecting to Wi-Fi so that it can later send data to the Thingier.io dashboard. The test was conducted by connecting the NodeMCU to Wi-Fi, and its status was monitored on a 16x2 LCD display. The NodeMCU test results are shown in Table 2.

**Table 2.** NodeMCU Testing Results

No	Test	Scenario	Result
1	NodeMCU booting and connecting to Wi-Fi	Powering on the NodeMCU.	
2	NodeMCU connected to Wi-Fi	Powering on the NodeMCU and the Wi-Fi router.	
3	NodeMCU failed to connect to Wi-Fi and created an Access Point	Powering on the NodeMCU while the Wi-Fi router is off.	

The table above shows the results of testing the NodeMCU's connection to Wi-Fi. The testing indicates that the NodeMCU is in good condition for use and can connect to Wi-Fi properly. It displays the status "connecting" when attempting to connect to Wi-Fi, shows the status "connected" when successfully connected, and creates an Access Point if it fails to connect to Wi-Fi.

### 3.4 DS18B20 Temperature Sensor Test Results

The DS18B20 temperature sensor test aims to measure the sensor's ability to detect water temperature. The sensor is placed in the center of the aquarium to ensure more accurate temperature measurements that represent the overall conditions of the aquarium, allowing the control system to maintain the temperature evenly. In this test,

the DS18B20 sensor is compared to a digital thermometer to calculate its accuracy, with the goal of ensuring the sensor functions properly. Accuracy is measured using MAPE (Mean Absolute Percent Error), calculated through equation (1) for percentage error and equation (2) for the MAPE value.

$$PE = \left( \frac{\text{Actual value} - \text{forecast value}}{\text{Actual value}} \right) \times 100 \quad (1)$$

$$MAPE = \frac{\sum \text{percentage error}}{n} \quad (2)$$

where MAPE is the Mean Absolute Percentage Error, PE is the Percentage Error, and  $n$  is the number of data points or sample size [7].

Testing at a water temperature of 25.3°C on the digital thermometer was followed by the following calculation:

Water temperature on the digital thermometer = 25.3°C

Water temperature on the DS18B20 sensor = 25.0°C

Percentage error =  $\frac{|25.3 - 25.0|}{25.3} \times 100 = 1.18 \%$

Using the same calculation method for the subsequent tests, the results of the absolute error percentage calculations are shown in Table 3 below.

**Table 3.** DS18B20 Temperature Sensor Test Results

Test	Measurement by Digital Thermometer (°C)	Measurement by DS18B20 Temperature Sensor (°C)	Error (%)
1	25.3	25.0	1,18
2	25.3	25.0	1.18
3	25.3	25.0	1.18
4	25.4	25.0	1.57
5	25.3	25.0	1.18
6	25.3	25.0	1.18
7	25.2	25.0	0.79
8	25.2	25.0	0.79
9	25.2	25.0	0.79
10	25,2	25.0	0.79

From Table 3, the Mean Absolute Percentage Error (MAPE) can be calculated as follows:

$$\begin{aligned} MAPE &= \frac{\sum \text{percentage error}}{n} \\ &= \frac{1,18 + 1,18 + 1,18 + 1,57 + 1,18 + 1,18 + 0,79 + 0,79 + 0,79 + 0,79}{10} \\ &= \frac{10,24}{10} \\ &= 1.02 \% \end{aligned}$$

Based on Table 3, the test results for the DS18B20 water temperature sensor module show that the Mean Absolute Percentage Error (MAPE) is 1.02%. This indicates that the readings from the DS18B20 sensor module fall into the "very good" category according to the classification in Table 4 for MAPE values.

**Table 4.** Classification of MAPE Value [8]

MAPE Value	Classification
< 10%	Very Good
10% ~ < 20%	Good
20% ~ < 50%	Usable
≥ 50%	Not Usable

### 3.5 Relay Testing

The relay testing, which is connected to the NodeMCU along with the RTC DS3231 and DS18B20 temperature sensor, aims to determine whether the relay can activate the fan relay, heater relay, and light relay. The relay setpoints are configured as follows: when the temperature is  $\geq 28^{\circ}\text{C}$ , the fan relay will turn on; when the temperature is  $< 25^{\circ}\text{C}$ , the heater relay will turn on. Additionally, the light relay will activate between 7:00 AM and 12:59 PM WITA and from 4:00 PM to 9:59 PM WITA. Outside of these times, the light relay will turn off. In the fan relay testing, the researcher conducted the test with water temperatures  $> 28^{\circ}\text{C}$ . The results of the fan relay testing can be seen in Table 5.

**Table 5.** Fan Relay Testing

Test	Water Temperature ( $^{\circ}\text{C}$ )	Relay Fan (ON/OFF)	Success (Matched /Unmatched)
1	28.19	ON	Matched
2	28.19	ON	Matched
3	28.13	ON	Matched
4	28.13	ON	Matched
5	28.13	ON	Matched
6	28.13	ON	Matched
7	28.13	ON	Matched
8	28.06	ON	Matched
9	28.06	ON	Matched
10	27.63	OFF	Matched

In the heater relay testing, the researcher conducted a simulation using water with a temperature  $< 25^{\circ}\text{C}$ . The results of the heater relay testing can be seen in Table 6.

**Table 6.** Heater Relay Testing

Test	Water Temperature (°C)	Relay Heater (ON/OFF)	Success (Matched /Unmatched)
1	23.88	ON	Matched
2	24.06	ON	Matched
3	24.13	ON	Matched
4	24.25	ON	Matched
5	24.38	ON	Matched
6	24.44	ON	Matched
7	24.56	ON	Matched
8	24.56	ON	Matched
9	24.69	ON	Matched
10	25.13	OFF	Matched

The results of the light relay testing can be seen in Table 7 below.

**Table 7.** Light Relay Testing

Time (WITA)	Relay Lamp (ON/OFF)	Success (Matched /Unmatched)
07.00	ON	Matched
13.00	OFF	Matched
16.00	ON	Matched
22.00	OFF	Matched

Based on the results from Table 5 for the fan relay testing, it is observed that when the temperature is  $> 28^{\circ}\text{C}$ , the fan relay activates. In Table 6 for the heater relay testing, it is evident that when the temperature is  $< 25^{\circ}\text{C}$ , the heater relay turns on. Table 7 for the light relay testing shows that the light relay is active between 7:00 AM and 12:59 PM WITA, as well as from 4:00 PM to 9:59 PM WITA outside of these hours, the light relay turns off. The results from the testing tables indicate that the relays operate as expected according to the predetermined setpoints.

### 3.6 Servo Motor Testing

The servo motor testing is conducted to ensure that the lever used for opening and closing the feeding container operates correctly. The output diameter is 1 cm. The testing begins at a  $10^{\circ}$  angle when the lever is in the closed position, and the servo's opening angle is adjusted to maximize the feed release. The results of the servo motor testing are shown in Table 8.

**Table 8.** Servo Motor Testing

Test No.	Servo Angle (°)	Feed Container Opening Condition (Open/Closed)	Width of Feed Opening (cm)
1	$10^{\circ}$	Closed	0 cm
2	$20^{\circ}$	Closed	0 cm
3	$30^{\circ}$	Closed	0 cm

Test No.	Servo Angle (°)	Feed Container Opening Condition (Open/Closed)	Width of Feed Opening (cm)
4	40°	Closed	0 cm
5	50°	Open	0.1 cm
6	60°	Open	0.3 cm
7	70°	Open	0.4 cm
8	80°	Open	0.8 cm
9	90°	Open	1 cm

From the results in Table 4.6, at angles of 10°, 20°, 30°, and 40°, the feeding container does not open sufficiently for feed release, as the output remains closed. It is observed that the feeding container can open to a maximum of 1 cm at a servo opening angle of 90°, which matches the size of the feed opening.

**Testing Recommended Delay Duration for Servo Opening with Respect to Feed Release and Duration of Fish Consumption.** The testing of the recommended delay duration for servo opening aims to determine the amount of feed released from the system to prevent excessive dispensing. The amount of feed released from the container depends on the angle and delay of the servo motor opening. The weight of the released feed was tested with delays repeated three times each at 0.1 seconds, 0.2 seconds, and 0.3 seconds. The results of the delay testing for the weight of the released feed can be seen in Table 9.

**Table 9.** Delay Testing for Released Feed Weight

Test	Feeding Action	Servo Delay (seconds)	Total Feed Dispensed (grams)
1	First Feeding	0.1	0.5
	Second Feeding	0.1	
2	First Feeding	0.1	0.5
	Second Feeding	0.1	
3	First Feeding	0.1	0.5
	Second Feeding	0.1	
4	First Feeding	0.2	1.3
	Second Feeding	0.2	
5	First Feeding	0.2	1.3
	Second Feeding	0.2	
6	First Feeding	0.2	1.3
	Second Feeding	0.2	
7	First Feeding	0.3	3.2
	Second Feeding	0.3	
8	First Feeding	0.3	3.2
	Second Feeding	0.3	
9	First Feeding	0.3	3.2
	Second Feeding	0.3	

The results of the recommended servo delay testing for feed release in Table 4.7 show that at a delay of 0.1 seconds, the total amount of feed released is 0.5 grams; at a delay of 0.2 seconds, it is 1.3 grams; and at a delay of 0.3 seconds, it is 3.2 grams. Subsequently, a test was conducted to determine how long it takes for the fish to consume the feed, with a total of 5 fish in the aquarium. The amount of feed needed and consumed by the fish can be seen in Table 10.

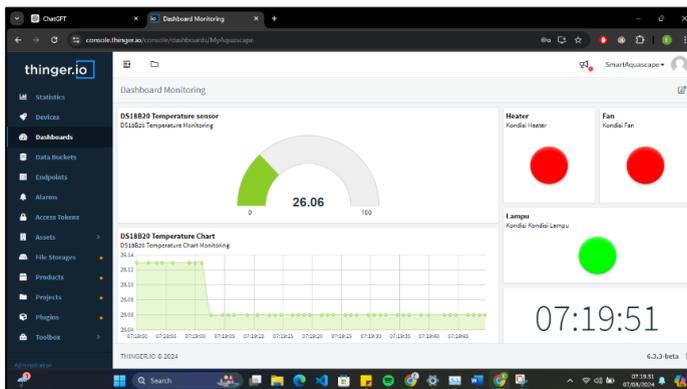
**Table 10.** Fish Consumption Duration Testing

Test	Feeding Schedule	Servo Delay (seconds)	Total Feed per Day (grams)	Fish Feeding Duration	Feed Condition (Finished/Leftover)
1	08.00 WITA 17.00 WITA	0.1 0.1	0.5	2 minute 52 seconds	Finished
2	08.00 WITA 17.00 WITA	0.2 0.2	1.3	3 minute 48 seconds	Leftover
3	08.00 WITA 17.00 WITA	0.3 0.3	3.2	4 minute 6 seconds	Leftover

In Table 10, the fish consumption duration testing shows that with a delay of 0.2 seconds, the total feed released is 1.3 grams, and with a delay of 0.3 seconds, the total feed released is 3.2 grams, which the fish cannot fully consume. However, with a delay of 0.1 seconds, where the total feed released is 0.5 grams, the fish can consume all the feed without any leftover. The results indicate that the fish can consume a total of 0.5 grams of feed in a day, taking 2 minutes and 52 seconds to do so.

### 3.7 Dashboard Thinger.io Testing Results

The Thinger.io dashboard testing was conducted to evaluate its capability to display measurement results from the sensors used in this research. The results of the testing can be seen in Fig. 8 below.



**Fig. 8.** Thinger.io Dashboard Testing

The results of the Thingier.io dashboard testing shown in Fig 3 display the temperature measurements from the DS18B20 sensor and the status of the relays in the device. Based on the testing, it is confirmed that the Thingier.io dashboard is capable of presenting the temperature data read by the DS18B20 sensor and the relay conditions. A green color indicates that the relay is on, while a red color indicates that the relay is off. Therefore, it can be concluded that the Thingier.io dashboard used in this research functions effectively in displaying measurement data from the sensors and the relay conditions utilized in the device.

### 3.8 LCD 16 x 2 Testing Results

The testing of the LCD 16 x 2 circuit was conducted to evaluate its ability to display time and measurement results from the sensors used in this research. The results of the testing can be seen in Fig. 9 below.



**Fig. 9.** LCD 16x2 Testing Results

Based on the results from the LCD 16 x 2 testing shown in Fig 4, it was found that the LCD 16 x 2 can display data in both letter and numeric formats. Additionally, it is capable of showing temperature readings from the DS18B20 sensor and the time from the RTC DS3231. In this research, the LCD 16 x 2 uses an additional I2C (Inter Integrated Circuit) module to conserve I/O pin usage. The serial communication between the I2C module and the NodeMCU uses the SDA and SCL pins, with SCL connected to pin D1 and SDA to pin D2 on the NodeMCU. Thus, it can be concluded that the LCD 16 x 2 circuit used in this research operates effectively in displaying time and measurement data from the sensors employed.

### 3.9 RTC DS3231 Testing Results

The RTC DS3231 testing was conducted to verify the accuracy of the time displayed by the RTC DS3231 compared to the actual current time. The comparison was made by observing the time output from the RTC DS3231 on the serial monitor and the real-time clock on the laptop, which was updated through an NTP (Network Time Protocol) server. The results of the RTC DS3231 testing can be seen in Table 11.

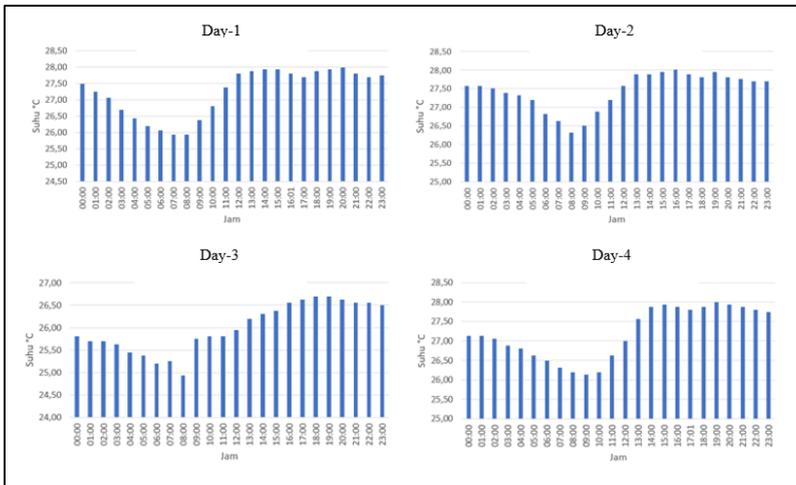
**Table 11.** RTC DS3231 Testing Results

No	Real-time Clock (Hours, Minutes, Seconds)	DS3231 RTC Time (Hours, Minutes, Seconds)	Difference
1.	07.58.43	07.58.41	2 Seconds
2.	07.58.44	07.58.42	2 Seconds
3.	07.58.45	07.58.43	2 Seconds
4.	07.58.46	07.58.44	2 Seconds
5.	07.58.47	07.58.45	2 Seconds
6.	07.58.48	07.58.46	2 Seconds
7.	07.58.49	07.58.48	1 Seconds
8.	07.58.50	07.58.49	1 Seconds
9.	07.58.51	07.58.50	1 Seconds
10.	07.58.52	07.58.51	1 Seconds

The table above presents the results from the RTC DS3231 component testing. The testing indicates that the RTC DS3231 functions well, with a minimal time difference between the RTC DS3231 and the real-time clock.

### 3.10 Overall System Observation of Temperature Control, Lighting, Automatic Feeder, and Monitoring Dashboard

The overall system observation was carried out to ensure that all components functioned as planned. This observation lasted for four days, with data being collected every hour throughout the day. This time frame was chosen because the water temperature in the aquascape tends to fluctuate.

**Fig. 10.** Temperature control observation

Based on the temperature control observation results in Fig.10, it can be concluded that the temperature control is functioning properly. The control process is activated when the temperature exceeds the predetermined setpoint. The temperature control is based on the water temperature readings in the aquascape, which are obtained from the DS18B20 temperature sensor.

**Table 12.** Lighting control schedule results

<b>Observation</b>	<b>Feeding Time</b>	<b>Servo (ON/OFF)</b>
1	07.00 - 12.59 WITA	ON
	13.00 - 15.59 WITA	OFF
	16.00 - 21.59 WITA	ON
	22.00 - 06.59 WITA	OFF
2	07.00 - 12.59 WITA	ON
	13.00 - 15.59 WITA	OFF
	16.00 - 21.59 WITA	ON
	22.00 - 06.59 WITA	OFF
3	07.00 - 12.59 WITA	ON
	13.00 - 15.59 WITA	OFF
	16.00 - 21.59 WITA	ON
	22.00 - 06.59 WITA	OFF
4	07.00 - 12.59 WITA	ON
	13.00 - 15.59 WITA	OFF
	16.00 - 21.59 WITA	ON
	22.00 - 06.59 WITA	OFF

The lighting control observation on the aquascape showed that the system is functioning properly, as seen in Table 4.12. The lighting is automatically set to turn on from 07:00 WITA to 12:59 WITA, then the lights turn off from 13:00 WITA to 15:59 WITA. After that, the lights turn back on from 16:00 WITA to 21:59 WITA and will turn off again at 22:00 WITA until the next day at 07:00 WITA. This setup ensures that the aquascape receives sufficient lighting, which is 12 hours per day.

**Table 13.** Feeding Schedule Observation Results

<b>Observation</b>	<b>Feeding Time</b>	<b>Servo (ON/OFF)</b>
1	08.00 WITA	ON
	17.00 WITA	ON
2	08.00 WITA	ON
	17.00 WITA	ON
3	08.00 WITA	ON
	17.00 WITA	ON
4	08.00 WITA	ON
	17.00 WITA	ON

Based on the feeding schedule observation results in Table 13, the system operates as scheduled. The automatic fish feeding is carried out by a servo motor that opens and

closes the feed container at predetermined times. The feeding schedule is set for two main times: 08:00 WITA in the morning and 17:00 WITA in the afternoon. This setup ensures that the fish receive a sufficient and regular amount of food, avoiding overfeeding or underfeeding.

The Thingier.io dashboard successfully displays temperature readings from the DS18B20 sensor along with a temperature chart, allowing users to view the history of temperature fluctuations. This monitoring system is designed based on the Internet of Things, enabling users to remotely monitor the temperature and device status by accessing the Thingier.io website. This provides convenience and peace of mind for users, as the system can not only be controlled automatically but also monitored from a distance.

From the system observation results, it can be concluded that all components of the device function properly as designed. This demonstrates that the system can effectively monitor and control the temperature, lighting, and fish feeding in an aquascape aquarium.

## 4 Conclusions

This system is designed using the ESP8266 microcontroller module, DS18B20 temperature sensor, a heater and fan for temperature control, and a 16x2 LCD for displaying temperature and time. Additionally, a servo motor is utilized to open the valve of the automatic feeding tube, while a Real-Time Clock (RTC) module serves as a time reference for the scheduling of lighting and feeding automation. The monitoring dashboard enables real-time temperature observation from a distance via the thingier.io website. As a result, aquascape owners can monitor water temperature anytime and anywhere, ensuring automatic control of temperature, lighting, and feeding to maintain an optimal environment for both fish and plants within the aquascape. The implementation results in the aquascape aquarium demonstrate that the performance test of the DS18B20 temperature sensor yields a Mean Absolute Percent Error (MAPE) of 1.02%. This indicates that the temperature readings from the DS18B20 sensor are highly accurate and can be effectively displayed on both the 16x2 LCD and the thingier.io monitoring dashboard. In the relay control test, the fan activates when the temperature reaches or exceeds 28°C, while the heater activates when the temperature falls below 25°C. The automated lighting control successfully operates, with the lights turning on from 07:00 WITA to 12:59 WITA and again from 16:00 WITA to 21:59 WITA, turning off outside these time ranges. The automated feeding system test shows that the fish in the aquarium are able to consume 0.5 grams of feed per day, with an average feeding duration of 2 minutes and 52 seconds.

## References

1. Warisno.: *Pesona Aquascape untuk Hobi dan Bisnis*. Lily Publisher, Yogyakarta (2011).
2. Raharjo, S.: *Sistem Otomatisasi Fotosintesis Buatan Pada Aquascape Berbasis Arduino*. KOMPUTEK 2(1) (2018).

3. Fernandi, A.: Mengenal Ikan Guppy: Jenis, Asal, dan Karakteristiknya, Cahaya Harapan Beo, Yogyakarta (2023).
4. Nopitasari.: Beternak Ikan Guppy Untuk Hobi dan Bisnis. Lembaga Kajian Profesi, Malang (2018).
5. Widjaja, T.: Aquascape: Pesona Taman dalam Akuarium. AgroMedia, Jakarta (2013).
6. Christofora K.: Mengenal Aquascape: Seni Mempercantik Akuarium Dengan Tanaman Hias, Edisi I. Cahaya Harapan, Yogyakarta (2023).
7. Yudaruddin, R.: Forecasting untuk Kegiatan Ekonomi dan Bisnis. RV Pustaka Horizon, Samarinda (2019).
8. Pulungan, J., Novita, R.: Prediksi Kebutuhan Energi Listrik Menggunakan Metode Jaringan Syaraf Tiruan. *Building of Informatics, Technology and Science (BITS)* 4(1), 294–302 (2022).

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

