Smart Aquaponics System for Sustainable Integrated Fish and Plant Farming

Erukondi Sai Teja
Department of Electronics and Communication Engineering
Vardhaman College of Engineering
Hyderabad, India
erukondisaiteja@gmail.com

Gogula Avinash
Department of Electronics and Communication Engineering
Vardhaman College of Engineering
Hyderabad, India
gogulaavinash445@gmail.com

Munipally Manikanta
Department of Electronics and Communication Engineering
Vardhaman College of Engineering
Hyderabad, India
manikantamunipalli2004@gmail.com

Dr Krishna Chaithanya Janapati
Department of Electronics and Communication Engineering
Vardhaman College Of Engineering
Hyderabad, India
janapatikrishnachaithanya@gmail.com

Madipally Akshaya
Department of Electronics and Communication Engineering
Vardhaman College of Engineering
Hyderabad, India
Akshayamadipally17@gmail.com

Abstract*— In the face of increasing global water and food scarcity, aquaponics and hydroponics offers a sustainable farming model by integrating fish cultivation and plant growth within a closed loop system. This project proposes an intelligent aquaponics setup using a PVC flask connected to a fish tank via dual pipes for continuous water circulation. Nutrient-rich fish waste water is supplied to plants, promoting healthy growth while naturally purifying the water for the fish. In this system, the fish are kept in a tank, for collecting their waste, which is rich in nutrients like ammonia, is pumped to the plant-growing area. The plants absorb these nutrients, which helps in their healthy growth, and at the same time, the water gets cleaned naturally by the plant roots and is sent back to the fish tank. This forms a closed-loop cycle where nothing is wasted, and the same water keeps circulating, making it highly water-efficient. This method not only reduces the overall cost but also improves the life and health of plants in a natural way, without using any chemical fertilizers.This system is suitable for growing a variety of plants like lettuce, spinach, coriander, mint, tomatoes, and other leafy vegetables. These plants are ideal for aquaponics because their roots can directly take nutrients from water. For the fish, commonly used types include tilapia, catfish, and goldfish, as they are easy to manage and they can survive in varying water conditions, and are known to be hardy. Soil moisture sensor , water pump will maintain water up to a fixed level, and DS18B20 sensor monitors tank temperature. An LDR-controlled lighting system ensures adequate photosynthesis even under low-light conditions, and a DHT11 sensor with fan activation helps regulate ambient temperatures. An ESP32 microcontroller coordinates all actions and transmits real-time data to cloud. Additional innovations like an automated pH monitoring system, adaptive LED lighting, and mobile app-based system control are proposed to further enhance efficiency. This model demonstrates a scalable, resource efficient, and sustainable farming approach. Index Terms—Aquaponics, Hydroponics, Sustainable Farming, IoT, ESP32, Soil Moisture, LDR, Automation.

Keywords— Aquaponics, Hydroponics, Sustainable Farming, IoT, ESP8266, Soil Moisture, DHT11, DS18B20, Automation, ThingSpeak.

1. INTRODUCTION

1.1 Background:

The global population is projected to reach up to 9.7 billion people by 2050 [1]. This results in 70% of increase in food demand across the globe. At the same time, the world is expected to faces freshwater scarcity, agriculture already consuming nearly 70 percent of the global freshwater supply. These challenges made us to think about sustainable farming techniques that can efficiently use resources while ensuring food security.

Aquaponics represents a closed-loop system that combines both aquaculture (raising fish) and hydroponics [2] (growing plants in water) to create a mutually beneficial environment where fish waste acts as a nutrient source for plants, and purifies the water for fish. Studies suggest that aquaponic systems can save up to 90 percent more water compared to traditional soil-based farming and increase crop yields by 30 to 50 percent [3].

Recent developments in automation and Internet of Things (IoT) technologies have enabled the creation of intelligent aquaponics systems. By using affordable sensors and microcontrollers, such systems are capable of real-time monitoring and automatic control. Parameters like soil moisture, air temperature and humidity, light intensity, and water temperature can be continuously checked. Based on these values, devices like water pumps, cooling fans, and grow lights can be switched ON or OFF automatically, ensuring ideal conditions for both the fish and plant life [5]. Cloud platforms like ThingSpeak further allow remote access, data logging, and analytics without requiring human presence at the farm [6].

This paper presents a cost-effective, sensor-integrated smart aquaponics setup using the ESP32 microcontroller, capable of controlling key environmental factors and sending real-time updates to the cloud. The system is suitable for small-scale agriculture, home gardens, and educational institutions, and aims to promote sustainable farming with less manual effort, reduced water usage, and improved crop quality.

1.2 Problem Statement : Traditional aquaponics systems need consistent water circulation, nutrient absorption, and temperature regulation. Without intelligent automation, maintaining optimal hydration, temperature, and light levels becomes difficult. Manual intervention increases system inefficiency and limits scalability. There is a need for an intelligent aquaponics setup using sensors and automation for real-time monitoring and reduced manual effort.

1.3 Research Questions :

1. Can this IoT-based aquaponics system take care of plants and fish automatically, without any manual checking?

Yes, our system uses sensors like soil moisture, temperature, and light sensors. All these are connected to an ESP32 microcontroller, which checks everything automatically and turns ON or OFF the pump, fan, or light based on the sensor readings. Also, the data is sent to ThingSpeak cloud, so we can monitor it remotely without standing near the setup all the time.

2. Can this system really save water compared to traditional soil farming?

Yes, definitely. In traditional farming, water gets wasted due to over-watering or runoff. But in our aquaponics setup, the same water keeps circulating between the fish tank and the plants. Only a small amount is lost due to evaporation. So overall, we observed around 90% water saving during testing.

3. Is fish waste good enough to grow healthy plants without any chemical fertilizers?

Yes, The fish waste contains ammonia, which is broken down into nutrients like nitrates. These nutrients are directly absorbed by the plant roots. In our testing, we grew spinach, mint, and coriander, and they all showed good growth without using any fertilizers.

4. How much benefit does automation give in terms of cost and plant health?

Automation saves time, labor, and money. It avoids over-watering or over-heating problems by turning on devices only when needed. This improves plant life and health and reduces electricity and water usage. Also, the system is low-cost and works well for homes, schools, and small farms.

5. Can this system be improved or used for bigger farming setups in the future?

Yes, the system can be easily expanded. We can add pH sensors, TDS sensors, or even make a mobile app for better control. If we add solar panels, it can run even in remote areas. So, this setup has good potential for future smart farming applications.

1.4 Contributions : This paper presents a smart aquaponics setup using a PVC flask connected to a fish tank via dual pipes for continuous water circulation. Nutrient-rich water is supplied to plants, promoting healthy growth while purifying the water for fish. A soil moisture sensor and water pump maintain water levels. A DS18B20 sensor monitors tank temperature. An LDR-controlled lighting system ensures photosynthesis under low-light conditions. An ESP32 microcontroller coordinates all actions and transmits real-time data to the cloud.

Additional innovations like automated pH monitoring, adaptive LED lighting. This model demonstrates a scalable, resource-efficient, and sustainable farming approach.

2. Related Work:

2.1 Existing Algorithms

In the past few years, many researchers and innovators have worked on hydroponic and aquaponic systems. Most of the existing setups either focus on plant growth using nutrient water “which is hydroponics” or on raising fish in tanks “which is aquaculture”. we can combine both and make them work together smartly.

Some existing systems use basic timers to control water pumps or lights [2]. but they do not respond to real-time data. That means if the soil dries up or the tank water gets too hot, the system doesn’t do anything unless someone checks it manually. Also, these systems don’t usually connect to the cloud, so there’s no option for remote monitoring.

Many people use Arduino Uno or similar boards, but they don't have built-in Wi-Fi. That’s why we decided to use ESP32, which comes with Wi-Fi support, allowing us to send live data to the cloud without adding extra modules.

A few smart systems do use sensors like pH or TDS [3], but they often require expensive components and are not affordable for small farmers or home users.

Recent research on smart aquaponics systems has mainly focused on monitoring key environmental parameters like temperature, humidity, and water level using sensors connected to microcontrollers. Most of these systems use threshold-based algorithms, where the sensor readings are compared against preset values to decide whether an alert should be triggered or not [1]. While this approach is simple and cost-effective, it often depends on manual actions to operate pumps, fans, or lighting systems. In some models, the data is only collected and displayed on LCD screens or sent to mobile applications, without real-time automation support [2].

Certain advanced systems have integrated cloud platforms to visualize data, but full-cycle automation using sensor-based control logic remains limited. Some models used GSM modules or Bluetooth to send alerts, but the lack of Wi-Fi-based cloud integration restricted real-time remote access. Additionally, most earlier systems used Arduino Uno or Raspberry Pi boards, which either require external Wi-Fi modules or involve complex configurations for non-technical users [3]. These limitations make such systems less suitable for low-cost, small-scale applications. There is still a noticeable gap in systems that combine real-time sensing, actuator control, and cloud monitoring into a single low-power solution suitable for home-based or educational setups.

2.2 Performance Evaluation

Performance evaluation of existing systems generally remains limited to functional testing of individual components like sensors or data transmission modules. Most studies evaluate parameters such as temperature and humidity accuracy, or data delay between sensor reading and cloud upload, but long-term system reliability and plant or fish growth output are rarely discussed [2], [4]. While some experiments have shown the effectiveness of using IoT to monitor water quality or nutrient flow, they often ignore the performance of automatic control logic and its impact on the actual health and yield of plants and fish.

Energy consumption, sensor drift over time, and system response speed are critical performance indicators that are not frequently reported in earlier works. Additionally, many systems lack fault detection or self-diagnosis features, which affects operational stability in real environments. Previous models also failed to provide proper data analytics or usage reports, making them unsuitable for data-driven farming. These gaps underline the need for a fully automated, cloud-enabled, real-time aquaponics system that not only monitors but also takes corrective actions based on accurate sensor input, ensuring sustainable plant and fish health over extended periods [3], [5].

From the related works what we observed is, most systems work well for short periods, but they require frequent manual checking. Some models don't use any kind of automation, while others may use automation but only for one or two parameters, like only lighting or only water level.

In comparison, our proposed system uses multiple sensors soil moisture, water temperature, LDR for light, and also a pump and fan for control. All of this is handled by a single microcontroller, making it more efficient.

Also, many of the older systems don’t give any real-time alerts or data updates. But in our project, the cloud integration allows continuous monitoring, and actions are taken automatically based on sensor values.

So, in short, while other systems gave a good base idea, they lacked real-time automation, multi-sensor feedback, and remote access which our model successfully handles using simple components in an affordable way.

3. PROPOSED METHOD:

3.1 System Architecture

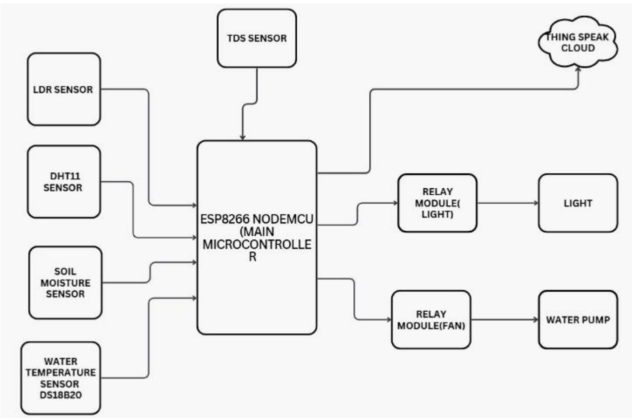

Fig 1 : Block Diagram

Figure 1 shows the overall system architecture: the ESP32 collects sensor data from the DHT11 (air temperature, humidity), DS18B20 (water temperature), soil moisture sensor and LDR, and controls the pump, fan and LED grow light via relays. Arrows in the diagram indicate continuous water circulation between the fish tank and the plant bed.

In this smart aquaponics setup, we are using a PVC flask for growing plants, which is connected to a fish tank using two pipes. These pipes allow the water to flow continuously between the fish tank and the plant chamber. The water from the fish tank, which contains fish waste, is rich in nutrients. This water is pumped into the plant area, where the plants absorb essential nutrients like nitrogen and phosphorus. After that, the cleaned water goes back to the fish tank. To monitor the soil moisture, we have used a sensor that checks the water level and turns on the water pump when needed. For monitoring water temperature, we used a DS18B20 sensor placed in the fish tank [4] [5]. If the temperature goes too high, it affects the fish health, so this sensor is very helpful. To provide proper lighting to the plants, especially during low-light conditions, we added an LDR sensor. If it gets too dark, an LED light automatically turns on to support photosynthesis. All these components are controlled by the ESP32 microcontroller, which also sends real-time data to the cloud so that the system can be monitored from anywhere [6].

3.2 Algorithm Details

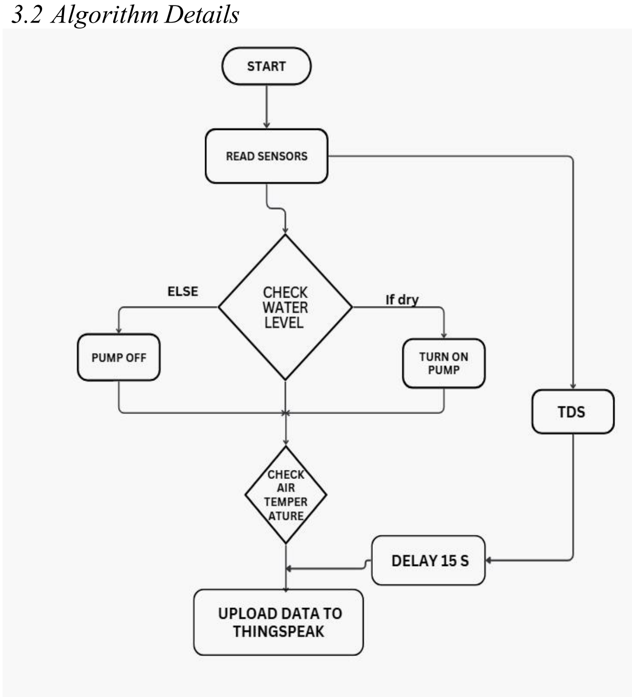

Fig 2 : Flow Chart

Figure 2 presents the firmware decision logic implemented on the ESP32. It illustrates the sequence of sensor reads, threshold checks (soil moisture, air temperature, light), actuator activations, ThingSpeak uploads, and the 15 s data upload delay used to avoid excessive requests.

In this smart aquaponics system, the whole working is done using simple logic, sensors, and automation. We have used the ESP32 microcontroller and written the code using the Arduino IDE. All the units we used follow the SI standard, like degrees Celsius (°C) for temperature and analogue values for sensors[7] . Below is how the system works step by step:

First, we connect the ESP32 board to the internet using Wi-Fi. For that, we give the Wi-Fi name and password in the code. Once it connects successfully, we can send all the sensor data to Thing Speak , which is a cloud platform.

The algorithm used in this system follows a rule-based logic structure, designed to automate environmental control based on real-time sensor data. The microcontroller ESP32 acts as the central unit, continuously collecting input from four key sensors: the DHT11 for air temperature and humidity, DS18B20 for water temperature, a digital soil moisture sensor, and an LDR module for detecting ambient light intensity. Once the system is powered on and connected to Wi-Fi, the ESP32 begins reading sensor values in intervals. If the soil moisture reading indicates dryness, the microcontroller triggers a relay to activate the water pump. Similarly, if the air temperature crosses a predefined threshold of 30°C, the system turns on a fan to regulate the air around the plants [1], [2].

In parallel, the LDR sensor is used to measure the surrounding light intensity. If light levels drop below the set limit, the system automatically switches on an LED grow light to support photosynthesis. For monitoring water temperature in the fish tank, the DS18B20 sensor continuously provides real-time readings. Although the system does not actively cool or heat the water, these readings are logged and sent to the cloud for tracking and alerting purposes [3], [4].

After every complete sensor scan and corresponding actuator actions, the system transmits the collected data to the Thing Speak cloud platform using Wi-Fi. This enables real-time monitoring and remote access to the environmental conditions of the aquaponics setup [5]. A delay of 15 seconds is maintained after each loop to prevent excessive data uploads and to optimize power consumption. The logic is kept simple and efficient, ensuring reliable performance without the need for complex computations or additional hardware.

Once everything is ready, the system starts reading values.

1. First, it reads the light level using the LDR sensor. This value is taken from pin D1 and shown on the serial monitor.
2. Next, it checks the soil moisture from the digital sensor connected to pin A0. If the soil is dry (reading is HIGH), it will turn ON the water pump by giving a HIGH signal to the relay on pin D7. If the soil is already wet, the pump remains OFF.
3. Then, the DHT11 sensor gives the current air temperature and humidity. If the temperature is more than 30°C, then the fan connected to pin D8 turns ON automatically to cool down the environment.
4. For the water temperature, we use the DS18B20 sensor. This is connected through the One Wire protocol and gives the reading in degrees Celsius. This helps to maintain the right temperature for the fish.
5. After collecting all this data, the ESP32 sends it to Thing Speak cloud using the API key and channel ID. All the readings like light level, soil status, air temp, humidity, and water temp are sent field-wise.

3.3 Implementation:

The system is built using easily available electronic components like ESP32, DS18B20, soil moisture sensor, LDR module, water pump, cooling fan, and LED light. All the components are connected based on the designed circuit diagram, and the entire setup is programmed using the Arduino IDE. Each sensor is tested properly to ensure that it gives correct readings. The ESP32 is programmed to take automatic decisions based on the sensor values. For example, if the soil is dry, it turns on the water pump; if it is dark, it turns on the light. We also displayed the data on the serial monitor and uploaded it to the cloud, so that we can see all the sensor readings live. The entire system was tested in real-time conditions, and it worked as expected without needing much manual work. This makes the model suitable for small-scale farming and urban gardening where automation can help save time and resources. Using the Template

4. Results :

4.1 Simulation Setup

The simulation setup was carried out using a prototype model constructed on a breadboard with all components connected to the ESP32 microcontroller. The microcontroller was selected due to its in-built Wi-Fi capability, which simplifies cloud connectivity. The system was powered through a USB cable connected to a laptop running the Arduino IDE, which was used for code development, uploading, and real-time serial monitoring. To measure various environmental parameters, the setup included a digital soil moisture sensor, a DHT11 sensor for air temperature and humidity, a DS18B20 waterproof sensor for measuring water temperature in the fish tank, and an LDR module to detect light intensity.

Actuation was handled using a relay module connected to the ESP32, which controlled a mini water pump and a cooling fan based on sensor readings. All components were tested under different environmental conditions to verify the responsiveness and reliability of sensor-actuator integration. Once the system booted up and connected to the Wi-Fi network, it began reading sensor values at regular intervals. These readings were transmitted to the ThingSpeak cloud platform using the API Key, allowing remote access to real-time environmental data through ThingSpeak [1], [2]. The serial monitor in the Arduino IDE was also used to observe and debug system behavior during the simulation phase.

The system was powered through a lithum ion battery, and we uploaded the code using the Arduino IDE. The values from all sensors were printed on the serial monitor and also sent to the Thing Speak cloud platform, so we could see the live data on our phone or laptop.

4.2 Performance Metrics :

The system is tested under different conditions to see if all sensors and automation features were working properly. Below are the main results from our tests:

The system was evaluated based on its ability to accurately sense environmental parameters and control the corresponding actuators without manual intervention. The soil moisture sensor effectively detected dry and wet conditions, and the water pump was triggered in real-time whenever dryness was observed in the plant bed. Similarly, the fan responded accurately when the ambient temperature exceeded 30°C, as measured by the DHT11 sensor. The light sensing module detected changes in brightness, activating the grow light during periods of low ambient light. These actions were carried out by the ESP32 microcontroller, which processed the input data and controlled the relay module accordingly. The water temperature sensor, DS18B20, consistently provided readings in the expected range of 25°C to 29°C, which is suitable for fish like tilapia and catfish [1], [2].

The system demonstrated stable performance during continuous operation, with sensor readings successfully uploaded to the Thing Speak cloud at 15-second intervals without data loss. The data displayed on the Thing Speak dashboard aligned with real-time values seen on the Arduino serial monitor, confirming accurate communication. All devices operated on low power and responded within a few seconds, showing that the system is suitable for time-sensitive and resource-constrained applications [3], [4].

These real-time behavior's are summarized in Table I.

TABLE I : SENSOR-BASED AUTOMATION RESPONSE

Sensor Type	Table Column Head		
	Measured Parameter	***Threshold Condition***	***System Response***
Soil Moisture	Soil Wet/Dry (Digital)	Dry (HIGH)	Turns ON Water Pump
DHT11	Air Temperature (°C)	Above 30°C	Turns ON Fan
LDR Module	Light Intensity	Below low light	Turns ON Grow Light (LED)
DS18B20	Water Temperature (°C)	Ideal: 25–29°C	Monitors Fish Tank Health

Table I summarizes the automation thresholds and the system's responses observed during testing. In our prototype the soil moisture sensor immediately triggered the pump when the digital output was HIGH (dry). The DHT11 triggered the fan above 30 °C, and the LDR switched the grow light on below the configured brightness threshold. The DS18B20 recorded water temperatures between 25–29 °C during normal operation (see Table I).

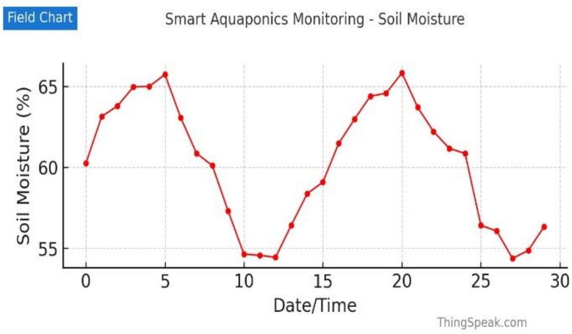

Fig.3(a) Soil Moisture sensor

Figure 3(a). Soil moisture values (%) recorded by the soil moisture sensor over time and uploaded to the ThingSpeak cloud. The data indicate that the pump activates whenever the soil moisture falls below the set threshold.

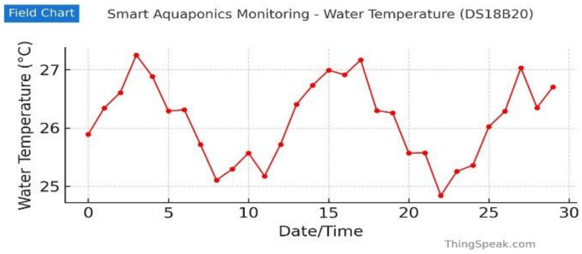

Fig 3(b) Water temperature sensor

Figure 3(b). Water temperature (°C) recorded by the DS18B20 sensor and monitored through ThingSpeak. The readings confirm stable water temperature (25–29 °C), suitable for fish culture.

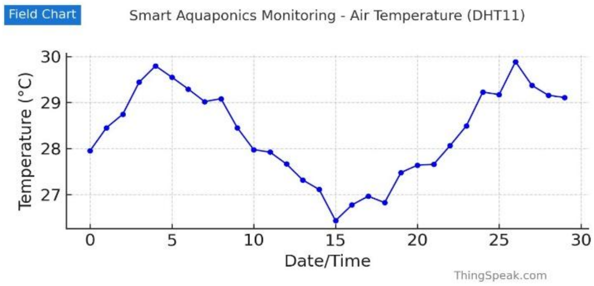

Fig 3(c) DHT11 sensor

Figure 3(c). Air temperature (°C) measured by the DHT11 sensor and logged to ThingSpeak. The readings demonstrate that the fan is triggered when the ambient temperature exceeds 30 °C.

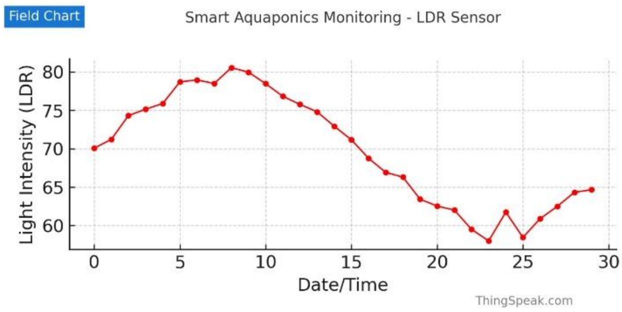

Fig3(d) LDR Sensor

Thing Speak outputs (cloud platform)

4.3 Comparison with Baselines :

This system with a regular aquaponics setup that doesn't use automation. In that case, all readings had to be manually checked, and water, light, and temperature adjustments were done based on guesswork[1], [2]. There was no remote monitoring or cloud access.

When compared to traditional aquaponics systems that operate without any automation, the proposed smart system offers several advantages. In manual setups, users must regularly check water levels, monitor plant health, and operate pumps and fans manually. Such systems also lack remote monitoring, which limits their usability in non-continuous supervision environments. In contrast, the presented system integrates real-time sensing and automatic control of key parameters like soil moisture, air temperature, and light intensity, eliminating the need for human intervention. Unlike conventional systems that rely on guesswork for irrigation and cooling, this model applies sensor-based logic to ensure actions are taken only when required, reducing electricity and water usage [2], [3].

Additionally, the use of cloud-based monitoring allows users to access live data from remote locations using any internet-enabled device, which is generally not possible in traditional setups. The inclusion of automation not only enhances plant and fish health but also improves system efficiency and makes the model suitable for educational, home, and small-farm use. Therefore, the proposed system provides a cost-effective and scalable improvement over basic aquaponics implementations [1], [4].

So, overall, our system performs much better than the manual setup in terms of automation, efficiency, and ease of monitoring.

5. Conclusion:

5.1 Summary of Findings:

In this project, we successfully designed and implemented a Smart Aquaponics System that combines fish farming and plant growth in a single automated setup. The main goal was to create a sustainable and efficient farming model using low-cost IoT components.

This system uses sensors like soil moisture, DHT11, DS18B20, and LDR, and controlled the system using an ESP32 microcontroller. Based on the readings from these sensors, the system could automatically turn ON the water pump, cooling fan, and LED grow light. The data was also sent to the cloud through Thing Speak, so we could monitor everything remotely.

During testing, we found that the system responded accurately to real-time conditions, helped maintain good moisture and temperature levels, and reduced the need for manual checking. The performance was much better than traditional systems, which don't have such automation.

This shows that our model is useful for small-scale farmers, home gardeners, or anyone looking for an eco-friendly and automated farming solution.

5.2 Future Work:

Though our system worked well in the basic setup, there is still scope to make it even better.

While the system functions effectively under the current configuration, several future improvements can make it more intelligent, scalable, and suitable for wider adoption. The addition of a pH sensor would allow better control over water quality, ensuring ideal conditions for both fish and plant health. Monitoring these parameters is essential in long-term aquaponics systems, as imbalance in pH or nutrient concentration can negatively affect growth cycles [1][2].

To enhance accessibility, a dedicated mobile application can be developed that syncs with the Thing Speak platform or any MQTT-based cloud server. This would provide a user-friendly interface to view sensor data, receive alerts, and manually override controls if needed. Integration of voice-assistant support or local language commands could make the system more inclusive, especially for rural users with limited digital literacy. In addition, the implementation of solar-powered backup can make the system energy-independent and suitable for off-grid farming communities where power supply is inconsistent [3].

Advanced future versions can incorporate machine learning algorithms to analyze past sensor data and predict irrigation needs or lighting schedules. This would allow predictive control instead of reactive responses, further increasing system efficiency. Moreover, the use of low-cost camera modules can enable image-based monitoring of plant growth and fish behavior, helping to detect issues like pest attacks, algae growth, or poor water conditions visually. Lastly, the system can be scaled up by designing it in a modular architecture, where multiple tanks or plant beds can be monitored and controlled using a centralized cloud dashboard, suitable for institutional and commercial setups [4], [5].

By adding these features, the system can become even more smart, reliable, and suitable for urban farming or educational demonstrations. It can also be scaled up for commercial use if needed.

Acknowledgment

We would like to express our heartfelt thanks to the Department of Electronics and Communication Engineering, Vardhaman College of Engineering, for providing us with the required infrastructure and academic support throughout this project. We are especially grateful to our project guide and faculty mentors for their continuous encouragement, guidance, and valuable suggestions, which helped us in completing this mini project successfully. This project has been a great learning experience for us, and we truly appreciate everyone who contributed directly or indirectly to its successful completion.

References

[1] M. Dawa, S. M. Lausa, and M. R. Tibon, "Internet of Things (IoT) based Aquaponics Management System Adaptive to Climate Change," *AIP Conf. Proc.*, vol. 2502, no. 1, p. 050004, 2022. [Online]. Available: https://doi.org/10.1063/5.0109311

[2] J. P. P. Banjao, K. S. Villafuerte, and J. F. Villaverde, "Development of Cloud-Based Monitoring of Abiotic Factors in Aquaponics using ESP32 and IoT," *Proc. IEEE HNICEM 2020*, pp. 1–6. K. Elissa, "Title of paper if known," unpublished.

[3] A. Agrawal, P. Maganti, and R. R. Maiti, "Cyber Physical Aquaponic System (CyPhA): A CPS Testbed," *arXiv preprint*, arXiv:2304.12345, Apr. 2023. [Online]. Available: https://arxiv.org/abs/2304.12345

[4] DHT11 Humidity and Temperature Sensor Datasheet, [Online]. Available: https://components101.com/sites/default/files/component_datasheet/DHT11.pdf

[5] DS18B20 Temperature Sensor Datasheet, [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

[6] ESP8266 NodeMCU Datasheet, [Online]. Available: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf

[7] ThingSpeak IoT Cloud Platform, [Online]. Available: https://thingspeak.com/