Scientific Papers. Series E. Land Reclamation, Earth Observation & Surveying, Environmental Engineering. Vol. XIV, 2025
Print ISSN 2285-6064, CD-ROM ISSN 2285-6072, Online ISSN 2393-5138, ISSN-L 2285-6064

INDOOR IoT SEEDLING NURSERY DEVELOPMENT

Peter UDVARDY[1], Matyas Csongor KOVACS[1], Levente DIMEN[2]

[1]Óbuda University, Alba Regia Technical Faculty, 45 Budai Street, 8000, Székesfehérvár, Hungary
[2]"1 Decembrie 1918" University of Alba Iulia,
5 Gabriel Bethlen Street, 510009, Alba Iulia, Romania

Corresponding author email: udvardy.peter@uni-obuda.hu

Abstract

In recent years, indoor plant cultivation has become an increasingly popular hobby. However, the natural environment required for optimal plant growth is not universally available, and therefore, it is necessary to replicate the missing environmental factors. Successful cultivation can be greatly enhanced by monitoring and adjusting environmental conditions. By collecting and analysing environmental data, it is possible to set appropriate parameters and optimize the operation of an environmental control system. This system requires sensors and actuators to compensate for environmental deficiencies and automate relevant processes. If the system's parameters are flexible enough, it can support the growth of a wide range of plant species. An effective approach to achieve this flexibility is through an online interface that allows remote access and enables the display of measured data as well as the adjustment of control parameters. A microcontroller-based system can be employed to manage this process. This paper outlines the hardware and software architecture, describes the communication protocols used, and provides an analysis of the system's expected performance.

Key words: *IoT, indoor planting, controlled environment.*

INTRODUCTION

Indoor plant cultivation

There are numerous reasons why individuals may choose to cultivate their own edible plants. While plant cultivation can certainly serve as a fulfilling hobby, extensive research has also demonstrated the positive effects of plants on human well-being. Moreover, various studies explore the specific impacts of different plant species on individuals (Yeo, 2021).

Another significant factor driving home cultivation is the rising cost of food, which is partly attributed to the increasing prices of imported goods. Such price fluctuations are influenced by global events beyond individual control. For instance, rising fuel costs lead to increased transportation expenses, which are ultimately passed on to consumers. In these situations, producing food locally can help reduce household expenditures, thereby mitigating some of the negative financial impacts. Consequently, the practice of growing at least locally suitable edible plants may regain popularity (Arora, 2019).

However, cultivating plants is not always a viable solution. Certain species may not be suitable for open-field cultivation in a given climate due to climate change or their inherent sensitivity to environmental conditions. Some plant varieties are highly susceptible to sudden heavy rainfall, which can damage their yield due to inefficient water distribution. Others struggle with prolonged drought periods, hindering their proper development.

A potential solution is to cultivate plants in controlled or semi-controlled environments, managing their development across various growth stages. Indoor cultivation methods have already gained traction, with industrial-scale operations successfully producing crops such as tomatoes in a cost-effective manner. However, such large-scale solutions are not feasible for the average individual, as not everyone has access to rooftop greenhouses or similar infrastructure. Instead, smaller, more portable, and user-friendly systems could provide a more practical alternative for everyday users (Pehin et al., 2020; Garcia, 2020).

In some cases, full environmental control is not required throughout the plant's entire lifecycle. Certain species only require additional support during specific growth phases, such as the early development stages when they are most vulnerable. This period typically includes germination and early seedling growth, lasting until the first buds appear. Afterward, some plants become sufficiently resilient to local environmental conditions, allowing for outdoor cultivation until the end of their lifecycle (Leavy & Hossain, 2014).

Environmental control processes must be executed periodically or under specific conditions in response to environmental fluctuations. However, maintaining such conditions requires a level of attention that many individuals, due to their busy schedules, may be unable to provide consistently. Additionally, some people may lack the expertise to recognize when intervention is necessary for optimal plant development. A common example is improper watering - either forgetting to water houseplants entirely or overwatering seedlings to the point of damaging or killing them.

Modern technology enables the automation of processes such as irrigation through the application of appropriate knowledge and tools. Numerous successful experiments and widely implemented systems already regulate or supplement specific environmental factors. Some advanced systems can simultaneously monitor and control multiple parameters, adapting to specific plant needs. However, complexity varies, and some existing solutions are unnecessarily intricate, reducing their efficiency (Palande et al., 2018).

The rise of IoT devices enables real-time monitoring and user adjustments. With intelligent algorithms and proper hardware, these systems can also become self-learning, making them easier to use and more efficient. Given these needs and technological advancements, designing a system that supports individuals in successfully cultivating plants at home presents a valuable and engaging research opportunity. Such a system could significantly improve accessibility and encourage more people to adopt home cultivation practices (Ragaveena et al., 2021).

MATERIALS AND METHODS

Lighting sensor

A wide range of lighting solutions is available for indoor plant cultivation, with their design significantly influenced by specific application requirements. One of the most critical characteristics of lighting fixtures is the electromagnetic wavelength spectrum in which they emit radiation.

For plants, the most suitable wavelengths for photosynthesis and photomorphogenesis - developmental processes influenced by light - are found within the visible light spectrum. Blue light (425-450 nm) and red-orange light (600-700 nm) play a crucial role in plant growth and development. Regarding ultraviolet (UV) radiation, wavelengths below 315 nm are considered harmful, as they can cause structural damage to plant cells.

Different regions of the visible light spectrum have varying effects on plant growth stages. In the early developmental phase, when stem and leaf formation are predominant, the blue wavelength range (400-500 nm) is the most effective, particularly for chlorophyll a absorption. In contrast, during the flowering stage, the red wavelength range (600-700 nm) becomes dominant, as it plays a crucial role in supporting chlorophyll b function (Paradiso & Proietti, 2022). Figure 1 shows the photosynthetic active light spectrum.

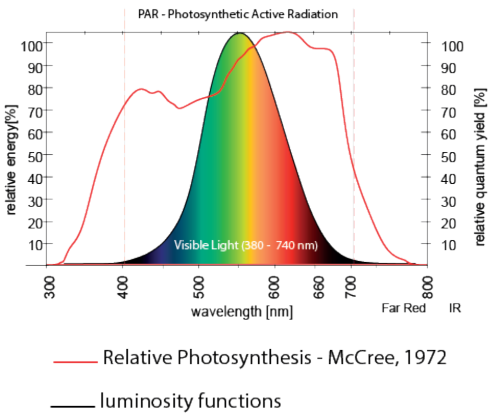

Figure 1. Photosynthetic active light
Source: https://hortione.com/2018/10/lighting-terminology-part-1-par-ppf-ppfd/

For successful indoor plant cultivation, it is essential to use light sources that emit radiation in the most suitable spectrum for each growth stage. During the vegetative phase - when the plant is growing but has not yet entered the flowering stage - the blue wavelength range (400-500 nm) promotes compact and robust growth. This reduces stem elongation, increases leaf density, and encourages branching, ultimately leading to higher flower production and improved yield (Beszédes et al., 2022).

Various lighting technologies are available for indoor plant cultivation, with LED-based solutions gaining increasing popularity over traditional incandescent bulbs and fluorescent tubes. LEDs are particularly favoured for their high efficiency, long lifespan, and customizable spectral output. In our own cultivation setup, natural light LED strips were used. Although I considered COB (Chip-on-Board) LEDs - known for their higher light intensity and broader coverage - I ultimately opted for lower-power LED strips. These offer easier installation, provide light tailored to the size of the cultivated area and specific plant needs, and require less cooling.

Soil Moisture Sensor

To optimize irrigation, it is essential to determine the moisture content of the soil in which plants are growing. This can be achieved using a capacitive soil moisture sensor, which, when connected to the analog input of an ESP32 microcontroller, represents soil capacitance variations as voltage levels. The capacitance increases proportionally with soil moisture content.

The sensor operates within a supply voltage range of 3.3 V to 5.5 V, allowing it to be powered directly from the 3.3 V output of the microcontroller. The core principle of capacitive sensing is that soil capacitance increases as moisture levels rise. The sensor utilizes a TL555C timer oscillator, which generates periodic square wave signals to charge the sensor. Higher capacitance results in longer charge times, leading to a decrease in the analogue signal read by the microcontroller.

Another approach to soil moisture measurement involves resistive sensors, which determine soil moisture based on electrical resistance. While these sensors are initially more cost-effective and simpler to implement, their major drawback is the oxidation of electrodes in direct contact with the soil, which shortens their lifespan. In contrast, capacitive sensors do not have exposed conductive surfaces in contact with the soil, making them more durable and capable of providing more reliable measurements over time (Mittelbach, 2012).

Other sensors

To determine the intensity of natural light, light-dependent resistors (LDRs) were applied, which vary their resistance based on the amount of incident light. For the measurement setup, I connected the LDR in series with a 10 kΩ resistor, forming a voltage divider. The voltage drop across the divider is measured by the analog input of an ESP32 microcontroller. The voltage value read by the analog input is directly proportional to the intensity of incoming light, enabling continuous monitoring of ambient illumination levels.

The HC-SR04 is an ultrasonic distance sensor widely used in various electronics and robotics projects for accurate and reliable distance measurement. It can be easily integrated with microcontrollers such as Arduino or ESP32 platforms. The HC-SR04 emits ultrasonic waves and detects the reflected signal. By measuring the time taken for the sound waves to travel to an object and return, the sensor calculates the distance between itself and the detected object.

The DHT22 (also known as AM2302) is a digital temperature and humidity sensor commonly used in various electronics and IoT projects for environmental data measurement. It is easy to use, provides accurate readings, and communicates with a microcontroller via a single data line (Yue et al., 2020).

Microcontroller

The ESP32 is more of a family of microcontrollers than a single microcontroller. It offers a wide range of variants that can be selected based on the specific application requirements and essential parameters. For my selection, I had access to three types of ESP32: the ESP32 S3, the ESP32 C6, and the "standard" ESP32 (such as the ESP32-WROOM or ESP32-WROVER modules). These models differ in several important aspects, primarily related to their architecture, functionalities, and intended application areas. The original ESP32 chip is widely available and popular, making it an ideal choice for general IoT projects such as sensor data processing, control tasks, or basic Wi-Fi and Bluetooth applications. It is particularly suitable for low-budget projects due to its costeffectiveness while still offering a robust set of features (Maier et al.,2017).

Software

Visual Studio Code, when complemented by PlatformIO, evolves into an open-source integrated development environment (IDE) that supports a wide range of microcontroller types. This integration provides an intuitive, yet professional development environment tailored for embedded systems engineers (https://code.visualstudio.com/).

VS Code is a highly versatile code editor widely adopted in the industry for programming in well-established languages such as Python, C++, C#, and many others. One of the key factors contributing to its widespread success is its open-source nature, which has led to the development of numerous extensions that allow users to tailor the editor to their specific needs. Among these extensions, PlatformIO stands out as a powerful open-source ecosystem designed for IoT system development. It integrates multiple development frameworks, maximizing the number of supported microcontrollers to ensure broad compatibility. The development frameworks supported by PlatformIO include Arduino, ESP-IDF: ESP32, Zephyr OS, Mbed OS, FreeRTOS and LibOpenCM3.

As seen from this list, PlatformIO fully supports ESP32 microcontrollers alongside numerous other platforms. Compared to the original Arduino IDE, the combination of VS Code and PlatformIO offers several advantages. However, due to its broad compatibility and extensive feature set, this platform is inherently more complex and requires a steeper learning curve than the traditional Arduino development environment.

The Arduino Integrated Development Environment (IDE) serves as a primary tool for programming Arduino microcontrollers, providing an intuitive interface for uploading programs to various types of microcontrollers. By default, the IDE offers native support for the Arduino platform.

In addition to its built-in capabilities, the Arduino IDE allows developers to extend its functionality through downloadable extensions, enabling the programming of microcontrollers from other manufacturers. One such extension provides support for ESP32-based microcontrollers, allowing developers to program Espressif-manufactured chips within the same development environment. The selected microcontroller's corresponding compiler processes the source code, ensuring compatibility with the target hardware.

DesignSpark Mechanical is a free, easy-to-use 3D Computer-Aided Design (CAD) software developed by RS Components, built on SpaceClaim technology. It is primarily designed for rapid prototyping, mechanical component and device modeling, and 3D printing projects. The software enables engineers and designers to quickly create and modify complex geometries, making it a valuable tool for product development and iterative design processes (https://www.rs-online.com/designspark/mechanical-software).

RESULTS AND DISCUSSIONS

System development process

In the system, an ESP8266 functioned as the web server and handled data transmission, while an Arduino Mega was responsible for data acquisition from sensors, actuator control, storing operational data in non-volatile memory, and providing a local menu interface to allow data input beyond the web interface.

The primary advantage of the dual-microcontroller system was the increased number of I/O pins, which facilitated the integration of additional sensors. However, based on empirical observations, the system provided more input and output pins than necessary. The ESP32 eliminates the need for two separate microcontrollers sharing tasks and exchanging data internally, thereby simplifying system architecture and reducing communication overhead.

First, the key data points were defined that the device should transmit via the web interface:

- soil moisture content expressed as a percentage, essential for irrigation control;
- threshold soil moisture level at which the system should automatically initiate irrigation;
- the direction of the highest light intensity and the light intensity itself, presented as a percentage. This information helps determine the primary natural light source

based on the system's orientation, which is particularly useful when operating in supplemental lighting mode;
- illumination activation time intervals, displayed in a tabular format;
- the current time according to the ESP32's internal clock;
- sensor scale limits, providing reference points for easier adjustment of new threshold values;
- a graphical representation of system-measured values over a specified time interval.

Next, it is essential to determine which parameters can be modified remotely via the web interface. Based on empirical observations, the following controls provide sufficient flexibility over the system:

- input fields for modifying the lighting activation time intervals;
- action button for changing the lighting operation mode;
- adjustable scaling for sensor measurements: To display sensor readings as percentages, a scale with defined upper and lower limits is required. Users can modify these limits via the web interface for both light intensity detection and soil moisture measurement through dedicated input fields;
- action button for switching the irrigation mode;
- adjustable thresholds for automated irrigation: Users can modify the automatic irrigation threshold via the web interface, as well as specify the "depth" of the water reservoir to define when it should be considered empty. These parameters can be set through dedicated input fields.

Ideally, when the ESP32 receives an empty request to its IP address (i.e., when there is no content following '/' in the URL), it automatically serves the web page to the client. This web page is stored as a string in the ESP32's memory and includes the current data managed within the microcontroller's variables. These data are transmitted along with the entire web page to the client.

When the user fills out a form and clicks the Submit button, the client sends the entered data to the ESP32 using the POST method. The GET method was deliberately avoided, as it appends data to the URL, leading to potential security and usability issues. For instance, data stored in the browser history or saved as a bookmark could be inadvertently reused, resulting in outdated or incorrect information being sent back to the web server. Figure 2 shows the ESP 32 data connection steps.

Figure 2. ESP 32 data connection

The POST method is a more secure and practical alternative, as the transmitted data do not appear in the URL or browsing history. The ESP32 executes the appropriate program function based solely on an action identifier associated with the URL (e.g., /submit1), which handles data reception and processing. After processing is completed, the ESP32 returns either an updated or a reset web page to the client, allowing the user to enter additional data or verify that the intended system changes have been successfully applied. The ESP32 initially stores the form-submitted data in its internal variables. If a change occurs, the data are subsequently saved to external non-volatile memory for long-term storage. Once the HTML code for the web page and the corresponding ESP32 code for serving the web interface are completed, the functionality of the web server can be tested. Figure 3 shows the flowchart of the main project programme.

The ESP32 must be connected to a network. To achieve this, the SSID (wireless network name) and password of the router are provided to the ESP32, allowing it to establish a connection. To determine the IP address assigned to the ESP32 by the router, one can either check the list of connected devices in the router's interface - provided that the router recognizes the ESP32's hostname - or identify the microcontroller using its MAC address. Some routers automatically assign a device name based on the ESP32's identifier, while others do not. For instance, on the router used during development, the ESP32 was assigned the device name ESP-283150. Alternatively, the microcontroller can be identified by retrieving its MAC address through other means.

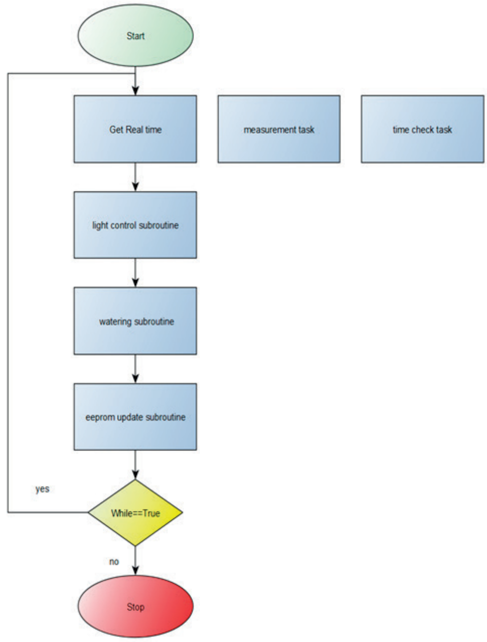

Figure 3. Flowchart of the main programme and tasks

IoT lighting system

The ESP microcontroller performs an hourly check to determine whether a predefined activation time requires the operation of the LED strips and whether a corresponding deactivation time mandates their shutdown.

If these conditions are met, the microcontroller activates the LED strips. The system allows only whole-hour time settings rather than minute-based configurations because the effect of illumination duration is negligible for time intervals smaller than an hour. However, if precise minute-level control becomes necessary, the program can be modified accordingly.

The lighting mode depends on the user-selected operating mode, in Supplementary Lighting Mode the duty cycle of the LED strips' PWM signal is adjusted based on ambient light intensity. This enables the system to optimize illumination by complementing natural light and in Illumination Mode the LED strips operate with a fully saturated PWM signal, meaning they emit continuous maximum brightness. Figure 4 shows how the light sensor reacts to the light intensity and light direction changes.

Due to their low heat emission, the LED strips do not require additional cooling. The generated heat is safely dissipated into the surrounding environment, preventing damage to both the LEDs and the supporting board during operation.

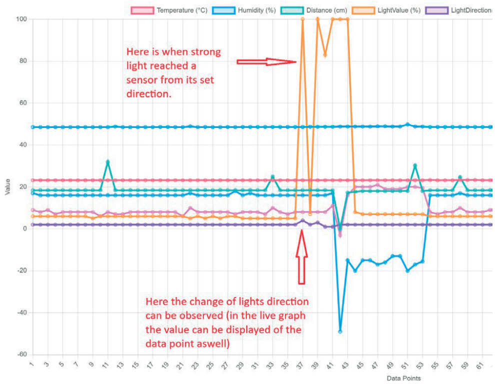

Figure 4. Changes in the light sensor readings/status

Soil moisture

The selected soil moisture sensors provide an output voltage ranging between 1.2 V and 2.8 V based on measurement results. These signals are converted into digital values by the analog input of the ESP32 microcontroller. Since the voltage range of individual sensors may vary slightly, these threshold values can be configured independently.

During the calibration process, the first step is to determine the sensor's output in dry air, which is typically around 3000. Next, by immersing the sensor’s soil-contacting portion in water, the lower threshold value is obtained, usually around 1000. After calibration, variations in soil moisture percentage can be accurately tracked, and the sensors provide reliable measurements.

Periodic recalibration may be necessary, for instance, when replacing sensors. The minimum and maximum values of new sensors can be easily set via the web interface. This feature is particularly useful because some newer sensor models, such as the capacitive soil moisture sensor v2.0, operate within a different voltage range (e.g., between 900 and 4000). This flexibility ensures that threshold values can be conveniently adjusted when replacing sensors.

During this process, only automatic irrigation is possible. Before executing irrigation, the system verifies whether there is a sufficient water level in the storage container; otherwise, the pump will not be activated. During irrigation, a predetermined volume of water is dispensed over a specific time period. The exact amount of water per cycle was determined through experimental testing. For the duration of irrigation, the lighting system is automatically turned off to indicate that irrigation is in progress and to allocate sufficient power to the pump, preventing system overload due to excessive current draw. After the irrigation cycle is completed, the lighting is reactivated if necessary.

If the size of the plant containers changes, it may become necessary to adjust the dispensed water volume or increase the number of irrigation cycles. Currently, these modifications can only be implemented by reprogramming the system. However, in the future, a more flexible and user-friendly solution could be developed if needed.

The system allows for two types of irrigation methods. As the name suggests, in top-down irrigation, water is applied to the soil surface and infiltrates downward under the influence of gravity. This method enables soil moisture sensors to respond immediately to irrigation since water reaches the sensor area more quickly. As a result, the measured moisture level rapidly reaches the predefined upper threshold. After irrigation, the soil moisture level gradually decreases, but the variation is not significant.

Figure 5 shows the flowchart of the irrigation management.

In bottom-up irrigation, water is delivered beneath the plant's soil, requiring it to be absorbed from below. This method is slower because water wicking through the soil takes time, particularly in larger volumes of substrate. Consequently, soil moisture sensors display delayed responses to irrigation. However, this can be advantageous, as it results in more stable moisture readings, avoiding abrupt fluctuations. Due to the slower response time, time-based control is necessary between irrigation cycles to prevent excessive water application. Additionally, this method requires sufficiently large overflow containers beneath the plant holders; otherwise, water may overflow. This issue can be managed by using adequately sized reservoirs. Although alternative solutions exist to prevent overflow, this method has proven to be effective in practice.

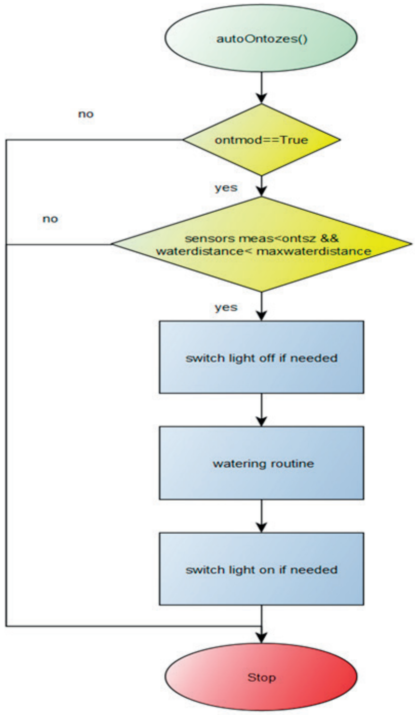

Figure 5. Flowchart of the irrigation management

Figure 6 shows the changes of soil moisture content and the threshold point.

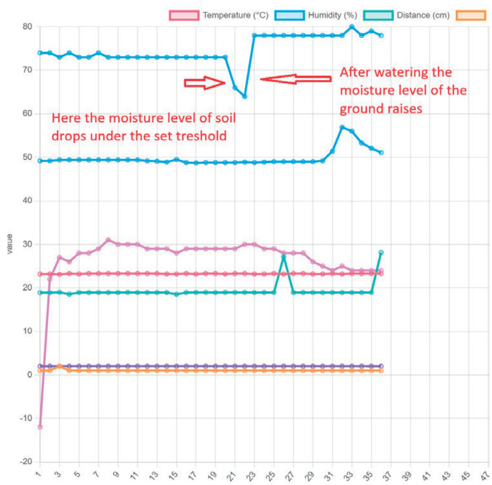

Figure 6. Changes of the moisture sensor status

CONSLUSIONS

The potential for further development of the system is somewhat limited due to the constraints of the ESP32 microcontroller. The primary limitation is its relatively low number of input and output pins. Additionally, certain pins cannot be fully utilized for all functions; for example, some internal resources required for ADC operations are also needed to support Wi-Fi functionality. As a result, ADC2 pins cannot be used as analog inputs within the system.

However, a more complex system could be designed by integrating multiple ESP32-based microcontrollers alongside a central processing unit. In such a configuration, individual units would handle specific tasks while the central unit would collect data via Bluetooth or Wi-Fi and manage the control of actuators based on the gathered information.

Since the LED strips are not fixed to the mounting surface, they can be easily replaced as needed, allowing for the use of strips with different light spectra. It is also possible to install multiple LED strips with varying spectra, and by modifying the program, users could select which strips should be active based on the current growth stage of the plants.

However, a drawback of this approach is that inactive strips do not emit any light, which may reduce the overall illumination output. On the other hand, if full illumination from all four strips is not necessary, and only half of them is sufficient, this can be advantageous. In this way, plants receive a light spectrum more closely aligned with their optimal growth requirements, ultimately improving cultivation quality.

If higher-power light sources, such as COB LEDs, are required, the system can accommodate their control. However, due to their significant heat generation, COB LEDs necessitate the use of large heat sinks to prevent overheating, which could damage the LEDs or substantially reduce their lifespan. For implementation, it is recommended to replace the current base that integrates the electronics and components with a substrate that can also function as a heat sink, as wood is not suitable for this purpose.

By controlling multiple pumps, it would be possible to irrigate the plants in individual seedling containers separately, significantly enhancing the system's flexibility and functionality. The ESP32 microcontroller has enough PWM-capable I/O pins to facilitate this implementation. Furthermore, the system could be improved by incorporating a predefined maximum soil moisture threshold during irrigation, preventing overwatering. This feature would be particularly beneficial for plants sensitive to excessive moisture, allowing the system to support the care of such species as well.

The system can be expanded with a camera module compatible with certain microcontrollers, such as the ESP32-CAM. By integrating an ESP32-CAM, it would be possible to visually monitor the plants, capture images, or even stream real-time video. The captured images could be displayed on the system's web interface, or the stream could be transmitted directly to the user's device. To implement this feature, in addition to the ESP32-CAM, a web server capable of handling video streaming would need to be developed. However, this integration would also increase the system's complexity, as the camera functions as a separate microcontroller unit.

An additional challenge may arise from the fact that transmitting the video stream to the central microcontroller and subsequently to the user client does not necessarily represent the most efficient use of resources. However, such an expansion could significantly enhance the system's functionality and user experience in the future.

Plant growth is influenced by various environmental factors, the monitoring of which can provide valuable insights for system users. For instance, soil temperature measurement can be achieved by integrating an appropriate sensor. However, expanding the current system is necessary, as all available analogue inputs are occupied. One possible solution is to integrate an additional microcontroller to manage the new sensor. Alternatively, a more extensive system redesign could be implemented, such as utilizing sensor modules that transmit data in digital format to the central unit via an interface like I2C. This approach would facilitate the connection of multiple sensors more efficiently.

The enhanced system offers a greater number of useful functionalities compared to its predecessors. Its efficiency has been improved, and its design has been simplified in certain aspects by utilizing a single microcontroller as its core, rather than two separate units with distinct functions. However, this design choice entails some limitations regarding the system's potential for further expansion. The system facilitates plant growth by automating key developmental processes and collecting data on both the plants and their surrounding environment.

Throughout the period in which the system supports plant growth, valuable data is gathered regarding the direction of optimal lighting exposure and other environmental factors that influence plant development. During the development process, multiple opportunities for further improvement were identified. These enhancements could significantly extend the system's functionality and lifespan while making plant care more convenient and efficient for users.

REFERENCES

Arora, N. K. (2019). Impact of climate change on agriculture production and its sustainable solutions. *Environmental Sustainability, 2*(2), 95–96. https://doi.org/10.1007/s42398-019-00078-w

Beszédes, B., et al. (2022). Artificial lighting experimental environment in agriculture for seed germination. In *Proceedings of AIS 2022 - 17th International Symposium on Applied Informatics and Related Areas* (pp. 77–81). Székesfehérvár: Óbudai Egyetem. ISBN: 9789634493020

Garcia, L. (2020). IoT-based smart irrigation systems: An overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture. *Sensors, 20*(4), 1042. https://doi.org/10.3390/s20041042

Leavy, J., & Hossain, N. (2014). Who wants to farm? Youth aspirations, opportunities and rising food prices. *IDS Working Papers, 2014*(439), 1–44. https://doi.org/10.1111/j.2040-0209.2014.00439.x

Maier, A., et al. (2017). Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. In *2017 Internet Technologies and Applications (ITA)* (pp. 143–148). IEEE. https://doi.org/10.1109/ITECHA.2017.8101926

Mittelbach, H., et al. (2012). Comparison of four soil moisture sensor types under field conditions in Switzerland. *Journal of Hydrology, 430–431*, 39–49. https://doi.org/10.1016/j.jhydrol.2012.01.041

Palande, V., Zaheer, A., & George, K. (2018). Fully automated hydroponic system for indoor plant growth. *Procedia Computer Science, 129*, 482–488. https://doi.org/10.1016/j.procs.2018.03.028

Paradiso, R., & Proietti, S. (2022). Light-quality manipulation to control plant growth and photomorphogenesis in greenhouse horticulture: The state of the art and the opportunities of modern LED systems. *Journal of Plant Growth Regulation, 41*, 742–780. https://doi.org/10.1007/s00344-021-10337-y

Pehin, S. F., et al. (2020). A study on youth aspiration and perception of agriculture and its policy implications. In A. J. Vasile (Ed.), *Handbook of Research on Agricultural Policy, Rural Development, and Entrepreneurship in Contemporary Economies* (pp. 441–453). IGI Global. https://doi.org/10.4018/978-1-5225-9837-4.ch022

Ragaveena, S., et al. (2021). Smart controlled environment agriculture methods: A holistic review. *Reviews in Environmental Science and Bio/Technology, 20*, 887–913. https://doi.org/10.1007/s11157-021-09591-z

Yeo, L. B. (2021). Psychological and physiological benefits of plants in the indoor environment: A mini and in-depth review. *International Journal of Built Environment and Sustainability, 8*(1), 57–67.

Yue, S. J., et al. (2020). IoT based automatic water level and electrical conductivity monitoring system. In *2020 IEEE 8th Conference on Systems, Process and Control (ICSPC)* (pp. 95–100). IEEE. https://doi.org/10.1109/ICSPC50992.2020.9305768

Microsoft (n.d.). *Visual Studio Code*. https://code.visualstudio.com/

RS Components (n.d.). *DesignSpark Mechanical Software*. https://www.rs-online.com/designspark/mechanical-software