Smart Control of Indoor Microgreens Growing Environment

Lucas Santos Pinow

Work oriented by:

Prof. Paulo Leitão

Prof. Claudio Leones Bazzi

Prof. Guilherme Luiz Moritz

Prof. José Barbosa

Bragança

2023-2024

Smart Control of Indoor Microgreens Growing Environment

Lucas Santos Pinow

Dissertation presented to the School of Technology and Management of Bragança to obtain the Master's Degree in Electrical and Computer Engineering under the scope of the Double Degree agreement with Federal University of Technology - Paraná.

Bragança

2023-2024

Dedication

Dedicate this work to God, to my family, and friends who supported me throughout, from beginning to end. Especially to my parents for always believing in me.

Acknowledgement

I thank everyone who has been directly and indirectly involved since the beginning of my technical/academic journey until today.

First and foremost, I would like to thank God for the opportunities, strength, and wisdom to deal with my decisions. I would also like to thank my family and friends, especially during this past year of distance.

I am grateful to my supervisors for always being willing to help me evolve and grow academically.

I would also like to thank IFPR for being the institution where I started in the world of electronics and automation. I am also grateful to UTFPR for the years I spent there, becoming and learning to be an engineer. And to IPB, for providing me with an international teaching experience that I never imagined witnessing.

Thank you all.

Abstract

Microgreens are young plants that have not gone through the full maturation period, resulting in smaller plants but rich in nutrients and flavors, making them highly recommended for restrictive diets focused on health and longevity. With the increasing demand for these products, solutions are emerging to supply the market; however, there are few options for domestic consumers to produce their own high-quality microgreens. Proper growth and maintenance of nutritional quality require an environment with good control of require an environment with good control of temperature, humidity, irrigation and lighting. This project aims to create an intelligent, automated, modular, and indoor greenhouse, allowing users to autonomously grow microgreens. Reference parameters are set based on recipes, featuring an efficient control system to manage the internal microclimate of the greenhouse, along with the integration of IoT technologies for data publishing and subscription for a monitoring system. As a test, basil was cultivated, taking 7 days to germinate and 21 days to develop, reaching a size of 2 cm. During the testing period, the average temperature was maintained at 21.7 °C during the day, with a setpoint of 20 °C, and soil moisture was kept at 77%, with a predefined minimum of 75%. The temperature control system proved effective when heating the environment, generating an oscillating temperature range around the setpoint, with a maximum variation of 0.3 °C.

Keywords: Microgreens, Automated Greenhouse, Control System, Microclimate, IoT

Resumo

Microverdes são plantas jovens que não completaram todo o período de maturação, resultando em uma planta menor, mas rica em nutrientes e sabores, sendo altamente recomendadas para dietas restritivas voltadas para saúde e longevidade. Com o aumento da demanda por esses produtos, surgem soluções para abastecer o mercado, porém há poucas opções para consumidores domésticos produzirem seus próprios microverdes com qualidade. O crescimento adequado e a manutenção da qualidade nutricional requerem um ambiente com bom controle de temperatura, umidade, irrigação e iluminação. Este projeto visa criar uma estufa automatizada, inteligente, modular e *indoor*, permitindo que o usuário cultive microverdes de forma autônoma. Os parâmetros de referência são definidos com base em receitas, contando com um sistema de controle eficiente para gerenciar o microclima interno da estufa, além da integração de tecnologias IoT para publicação e subscrição de dados para um sistema de monitoramento. Como teste, foi realizado o cultivo de manjericão, que levou 7 dias para germinar e 21 dias para se desenvolver, alcançando um tamanho de 2 cm. Durante o período de testes, a temperatura média foi mantida em 21,7 °C durante o dia, com o setpoint definido em 20 °C, e a umidade do solo foi mantida em 77%, sendo que o valor pré-definido era de no mínimo 75%. O sistema de controle de temperatura mostrou-se eficaz quando precisava aquecer o ambiente, gerando uma faixa de temperatura oscilante em torno da temperatura definida, com uma variação máxima de 0,3 °C.

Palavras-chave: Microverdes, Estufa Automatizada, Sistema de Controle, Microclima, IoT

Contents

List of Tables

List of Figures

List of Acronyms

3D Three-Dimensional. 23

EEPROM Erasable Programmable Read-Only Memory. 32

FR Functional Requirements. 15

GPIOs General Purpose Inputs/Outputs. 47

HMI Human-Machine Interface. 17, 20, 32

I^2C Inter-Integrated Circuit. 30, 32

IoT Internet of Things. 2, 10, 51

IPB Instituto Politécnico de Bragança. 27

JSON JavaScript Object Notation. 49

LCD Liquid Crystal Display. 32

MDF Medium Density Fiberboard. 27

MIMO Multiple Input Multiple Output. 8

MQTT Message Queuing Telemetry Transport. 12

NFR Non-Functional Requirements. 15

OLED Organic Light-Emitting Diode. 32

PA Precision Agriculture. 11

PI Proportional-Integral. 2

PID Proportional-Integral-Derivative. 2, 9, 58

PLC Programmable Logic Controller. 8

PTC Positive Temperature Coefficient. 20, 38

PWM Pulse Width Modulation. 38

RTC Real-Time Clock. 20, 32

SPI Serial Peripheral Interface. 30

USART Universal Synchronous Asynchronous Receiver Transmitter. 30

Wi-Fi Wireless Fidelity. 18

Chapter 1

Introduction

There is a notable increase in the global demand for organic and nutritious foods, especially those that adhere to the most restricted diets, with the aim of promoting health and longevity [1], [2]. There is also a growing awareness about the potential long-term adverse impacts associated with consuming genetically modified vegetables [3]. In this context, the rise of microgreens emerges as a viable food alternative for inclusion in the everyday diet. These are recognized as more nutritious and tasty food options compared to conventional plants [1], [2], [4].

Microgreens represent plants at a young stage, which have not completed all phases of their growth cycle, resulting in smaller dimensions and reduced maturity compared to their mature counterparts [5]. It is crucial to highlight the distinction between microgreens and sprouts, which are also harvested early. Most sprouts are grown in environments lacking adequate lighting and with excessive levels of soil moisture, creating conditions favorable to the development of unwanted microorganisms [1], [4]. These microorganisms can compromise the growth and performance of plants, in addition to representing a potential risk to the health of consumers [6]. On the other hand, microgreens are produced in controlled environments, in which the appropriate levels of lighting, temperature and humidity are provided for plant development [4].

Greenhouses, in general, improved the plant growth process, providing an environment of protection against atmospheric agents and creating an internal microclimate that favors the plant's physiological processes, in addition to improving its quality and yield when compared to the open field [7].

Automation in greenhouses has revolutionized modern agriculture, providing precise control of environmental conditions and significantly improving crop efficiency and productivity. Sensors and actuators connected to a processing unit with a control algorithm continuously adjust variables such as temperature, humidity, and lighting, creating an ideal environment for plant growth, regardless of external climate fluctuations. Commonly applied techniques in control include fuzzy logic and smart fuzzy logic, often associated with Proportional-Integral (PI) controllers and the classic Proportional-Integral-Derivative (PID). This results in healthier plants and higher quality harvests. Furthermore, automation allows for the efficient use of resources, such as water and nutrients, applying them precisely and avoiding waste, which reduces operational costs and the environmental impact of agricultural production [8], [9].

In this sense, the emphasis on the production of microgreens in greenhouses with precise control has grown significantly, seeking to establish an environment conducive to the development of plants [1]. One of the most promising approaches is the implementation of automated greenhouses with Internet of Things (IoT) solutions [8]. By allowing real-time monitoring of variables such as temperature, humidity and lighting, such solutions enable farmers to precisely adjust growing conditions, aiming to maximize crop productivity and quality. Furthermore, the remote monitoring capability offered by IoT solutions provides more flexible management of greenhouses, allowing farmers to monitor and control the cultivation process efficiently, regardless of their physical location [8], [10].

1.1 Justification

The growth in demand for healthy and organic foods has driven the search for more efficient and sustainable farming methods. Microgreens, with their high nutritional value and fast growth cycle, represent an excellent option to meet this demand [4]. However, growing microgreens in uncontrolled environments can result in significant variations in plant quality and yield, as well as increasing the risk of contamination by unwanted microorganisms [1].

The use of automated indoor greenhouses appears as a viable solution to these challenges, especially aimed at home cultivation. By strictly controlling environmental variables such as temperature, humidity and lighting, it is possible to create ideal conditions for the growth of microgreens, ensuring consistent, high-quality production [9], [11].

To meet domestic demand, the development of an automated greenhouse for microgreens production offers a practical and technological solution that can be easily integrated into residential environments. This approach not only facilitates consumer access to fresh, nutritious foods, but also promotes sustainable agricultural practices by enabling local cultivation and reducing dependence on transportation and logistics associated with traditional agricultural production [8], [9], [11].

1.2 Problem Statement

Automated greenhouses represent an effective solution for optimizing plant growth by providing a controlled environment that protects against adverse weather variables and pests [1], [8]. However, most commercially available automated greenhouse technologies are designed for large-scale agricultural operations. Commercial solutions offered by companies such as Argus Control Systems, Priva, and Autogrow provide advanced environmental control systems, but these solutions are often inaccessible or unsuitable for home use, where consumers seek efficient and convenient methods for growing plants in limited spaces and with restricted resources.

The lack of automated growing solutions suitable for the home environment limits the efficient and consistent production of plants, such as microgreens, for consumers. In this sense, there is a need to develop a solution that meets this demand, making the most of the limited space in residential environments. Additionally, there is a growing necessity to research, develop, and apply new control techniques that differ from conventional methods. These innovative approaches are crucial for overcoming current limitations and optimizing plant growth in constrained spaces.

This study aims to explore a solution to this problem, focusing on modularization, verticalization, autonomy, intelligence, versatility, and robustness. Modularity and verticalization are crucial for maximizing the use of small spaces, allowing users to grow a greater quantity of microgreens in limited areas [11]. Autonomy is essential for making accurate decisions in controlling the environment, creating a suitable microclimate for plant growth [11]. Intelligence, through IoT technology, enables remote monitoring, offering real-time control and adjustments [12]. Versatility allows the production of different types of microgreens, adapting to the specific needs of users. Robustness ensures that the system is reliable and durable, providing a sustainable and efficient solution for home microgreen production [11], [13]. Additionally, the study will explore the development of an efficient control system specific to each variable.

1.3 Objectives

Based on the contextualization of the problem and its importance, the main objective of this work is to develop an intelligent, indoor, and modular automated greenhouse intended for the production of microgreens for domestic consumption, with the capability of integrating with a remote monitoring system using IoT technology.

In this way, some specific details about the intended direction for the construction of this system can be defined. Thus, the general requirements of the intelligent, indoor, and modular automated greenhouse to be developed during this work are:

- Through the greenhouse control system, it must be possible to create a customizable environment where it is possible to control all environmental variables, such as temperature, humidity, soil moisture and lighting.

- The system must be capable of sending greenhouse data to a remote monitoring system in real time, and to receive commands to manipulate its actuators. Furthermore, it must be able to receive a recipe to be followed during the production of microgreens. This recipe must contain setpoints for temperature, humidity, soil moisture, light intensity, and light exposure time, divided between day and night.

1.4 Thesis Structure

This thesis is structured into six chapters that detail all the work developed.

Chapter 1 presents the introduction of the work and its objectives, offering a contextualization for the field of application, in addition to the reasons that justify carrying out this study.

Chapter 2 discusses the state of the art of existing technologies and solutions, including similar work in the areas of greenhouse automation, microgreen production and IoT technology for similar applications.

Chapter 3 describes the methodology adopted for developing the system, presenting all functional and non-functional requirements, in addition to presenting the system architecture.

Chapter 4 details the development of the system, from the physical construction of the indoor greenhouse prototype to the integration of the IoT perspective. Furthermore, this chapter describes how the control system was developed to adequately meet all established requirements.

Chapter 5 reports the tests and results obtained with the system, ranging from dry runs to the evaluation of a microgreen crop plantation.

Chapter 6 presents the conclusion of the tests and results obtained, evaluating the effectiveness of the prototype control, its limitations and possible improvements, in addition to suggesting directions for future work.

Chapter 2

State of Art

This chapter presents an overview of existing technologies and works that were developed with objectives similar to this study. Existing solutions, results and investigations in the microgreens, automated greenhouses and IoT applications will be also covered.

2.1 Automated Greenhouse

In Chapter 1, an introduction to automated greenhouses was presented. In this section, applications of automated and smart greenhouses will be addressed, focusing on current control systems and IoT technologies.

Bhujel et al. [14] carried out a study on the importance of strategically applying sensors in greenhouses, aiming to find a position that better reflects the plant's behavior in relation to the environment. They concluded that the sensors should always be above the height of the plants. Furthermore, the authors emphasized that future generations of greenhouse control and monitoring systems are moving towards intelligent, real-time, remotely accessible and fully automatic systems.

Jin et al. [15] developed a remote measurement and control system using GSM technology for communication between information systems, applied to large-scale greenhouses. The system consists of a microcontroller, LM-35D and CHM-01 temperature and humidity sensors, an illumination sensor, and a CO_2 sensor. Actuators such as a clearstory, shade, sprinkler, circulatory fan, and CO_2 generator were used to manipulate the environment. The system is controlled remotely via a base station.

Li et al. [16] present an intelligent control system for greenhouses, using a Programmable Logic Controller (PLC) and fuzzy logic to create a predictive control system. Temperature was measured using AD590 sensors, air humidity with HS1101 sensors, light intensity with HSTL-GZD sensors, and CO_2 levels with B-530 sensors. Actuators used include a wet curtain, wet curtain pump, fill lights, a fan, a heater, and a sun-shade net. The control logic applied for each variable involves direct comparison of measured values with reference values, followed by fuzzy logic decision-making for actuator operation. The preference for fuzzy logic was due to the system's Multiple Input Multiple Output (MIMO) nature and the wide variety of actuators employed. Therefore, the authors concluded that this system, operating at a higher level of complexity, provides a more stable environment for production. This technique still needs further exploration in the future.

Kang et al. [17] developed a remote control and monitoring system for a greenhouse using the ZigBee communication protocol, a pioneering technology in IoT. They employed sensors such as the SHT71 for temperature and humidity, the IRtec Rayomatic leaf temperature sensor for non-contact temperature measurement using infrared, and the Model 237 Leaf Wetness sensor for indirect leaf humidity measurement without contact with the plant. Actuators included windows, fans, and heaters. The MSP430 microcontroller from Texas Instruments was utilized for control. Sensor nodes were constructed with these sensors and a CC2420 RF 2.4GHz transmitter for ZigBee-based monitoring, while actuator nodes were implemented for environment control. The activation of actuator nodes is electronically managed using the 74LVC244 buffer and the 74HC08 AND gate, triggered upon receiving critical values from the sensor nodes.

Another application of microcontroller-based control systems is demonstrated in the work of Alyousif et al. [18], where they developed a cost-effective system for data collection and environmental control using fuzzy logic. They employed the DS1820 temperature sensor and the PIC16F877A microcontroller. Actuators such as red and blue LEDs, fans, and a heater were utilized for temperature control. The control algorithm integrates simple techniques with more sophisticated methods, combining PID control and fuzzy logic. It operates in two stages, initially, fuzzy logic acts swiftly to adjust the temperature close to the setpoint by comparing the measured temperature with the reference value, generating a temperature error (E) and its rate of change (EC). In the second stage, PID control is applied to minimize overshoot and static error based on E and EC values. However, the testing environment, including the control logic, was conducted solely in a simulated environment using MATLAB. This approach limited the validation to theoretical simulations rather than real-world implementation.

Wu et al. [19] developed an automated system using advanced IoT technologies and embedded platforms like Raspberry Pi 3B+ to enhance greenhouse control. Their system improved monitoring and control of soil pH through the use of fertilizers and PID control. Additionally, they employed E32-TTL-100 LoRa technology for long-distance data transmission. For controlling the greenhouse solution, they utilized four tanks with different solutions, an acid tank, a venturi, solenoid valves, a mixing tank, and EC/pH sensors, inlet pipe, and output pump. The Raspberry Pi was interconnected with these components, managing the opening of solution tanks based on EC/pH sensor readings through solenoid valves, directing the flow to the mixing tank. This project predominantly focused on substrate fertilization mixing compared to other variables typically addressed in related literature.

Li et al. [20] developed a simple temperature control system applied to a greenhouse, based on the ADT14, an integrated temperature control circuit. Using an output relay to activate a heater, they controlled the temperature through a hysteresis-based electronic circuit, with a temperature variation range of 2.5°C to 1.55°C around the setpoint. The circuit proved highly efficient due to its low-cost components, requiring only the ADT14 integrated circuit along with resistors and diodes to create a temperature control node. However, it has several limitations, particularly high-temperature variation in sensitive environments and the lack of integration capability with IoT technologies. Nevertheless, the control technique discussed is relatively simple to implement in microcontrollers, making it a viable alternative for temperature control and potentially other variables in a greenhouse.

In this work, the developed system will rely on a microcontroller that integrates sensors and actuators to achieve precise control. Specific algorithms and control techniques will be devised for each environmental variable.

2.2 Internet of Things

The IoT refers to the interconnection of devices and systems via the Internet, allowing them to collect, share and analyze data. This technology has a significant impact on several industries, including agriculture, where IoT has been used to optimize processes, improve efficiency and reduce costs. In the context of automated greenhouses and microgreen production, IoT offers several advantages, such as real-time monitoring, task automation and data-driven decision making [21].

IoT has been growing rapidly in many domains such as automotive, aviation, automation, energy and healthcare. However, one of the difficulties encountered is establishing a standard concept, since requirements may vary for each application. Recognizing the advantages of IoT in terms of variability of applications and applied technologies, the IoT taxonomy was presented [22].

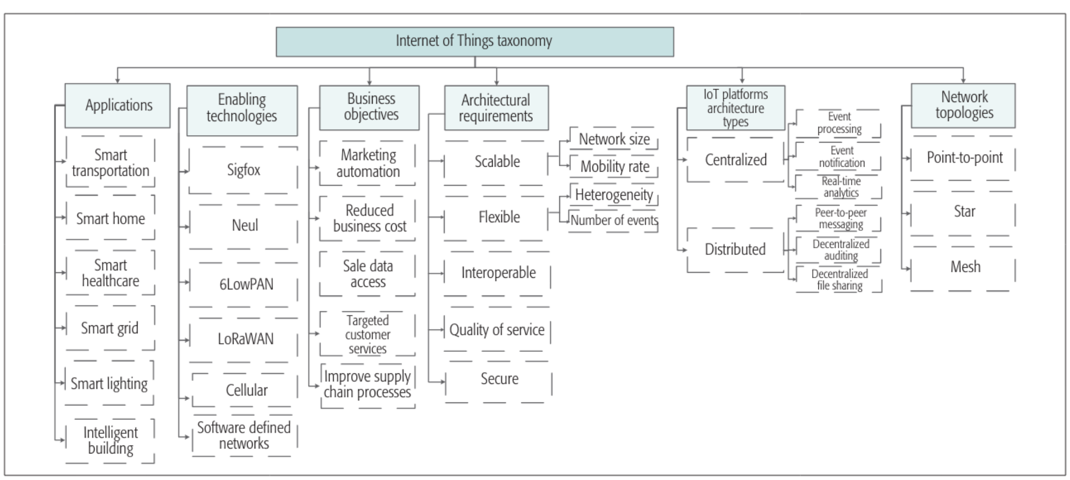


Figure 2.1: IoT Taxonomy [22].

Figure 2.1 shows the IoT taxonomy, providing a comprehensive view of this technology by presenting variability in applications, intrinsic technologies, business objectives, architectural requirements, in addition to indicating centralized or distributed platforms and network typologies. With this vision in mind, IoT applications can be described and analyzed more efficiently and directly.

One of the emerging applications of IoT is its use in Precision Agriculture (PA). This type of system is developed to improve processes such as precise monitoring and treatment. Using cloud-based services and IoT control centers for real-time data collection and processing, there is automation of environmental control, including the actual planting, fertilization and harvesting of crops, at the appropriate time and duration [23].

For precision agriculture, a possible information architecture for describing the process is represented by Figure 2.2.

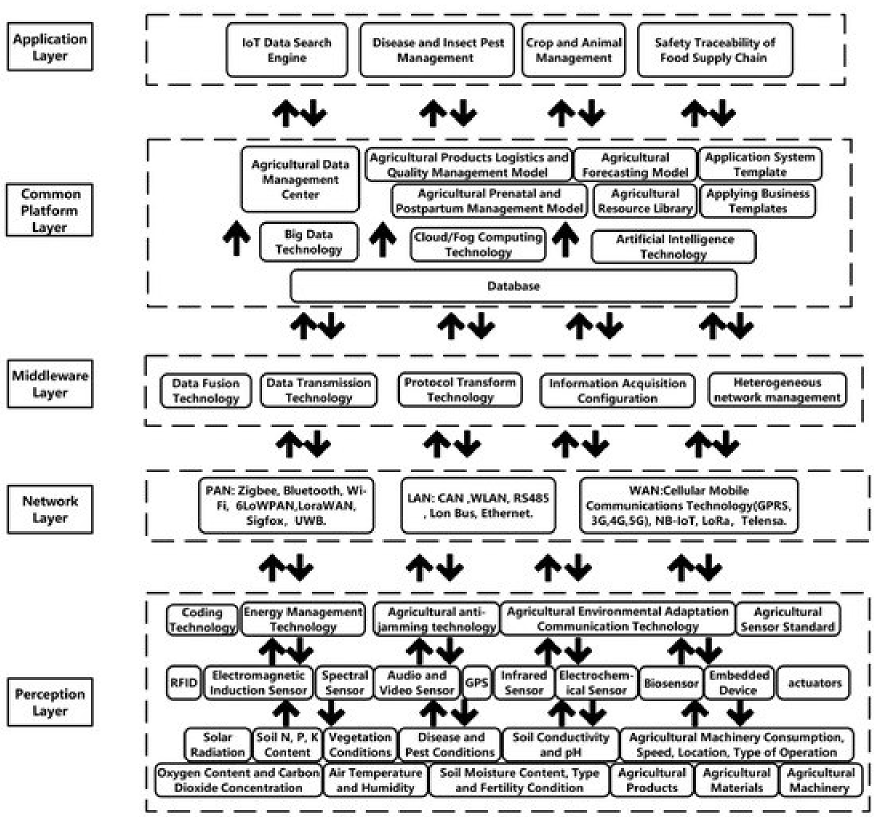


Figure 2.2: IoT Architecture for PA [24].

Figure 2.2 illustrates the relationship between layers and levels of information transmission in the context of IoT. The Perception layer is made up of various sensors, terminal devices, agricultural machinery and wireless networks such as WSN, RFID tags and readers. These sensors include environmental monitors, sensors for information about plant and animal life, fundamental to agriculture. They capture data such as temperature, humidity, wind speed, plant diseases, pests and animal health indicators. This information is initially processed by embedded devices before being transmitted through the Network layer, which serves as the core infrastructure of IoT. This layer combines several types of communication networks, including wired technologies such as CAN bus and RS485, and wireless technologies such as Zigbee, Bluetooth, LoRa and NB-IoT. In addition to forwarding agricultural data collected by the Perception layer, the Network layer also sends commands from the Application layer to control devices, ensuring specific actions [24].

Another approach to transmitting and publishing data can be using the Message Queuing Telemetry Transport (MQTT) protocol. This protocol, by default, operates with TCP/IP on port 1883 and has different implementations, such as Mosquitto, HiveMQ and Eclipse Paho MQTT. MQTT has become a favorite in the IoT field due to its publish/-subscribe model that offers simple and flexible implementation. Another advantage of MQTT is its scalability, where a single agent can publish data and a wide range of agents can receive this information with a simple implementation. Using a broker to manage data, MQTT has characteristics as illustrated in Figure 2.3 [25].

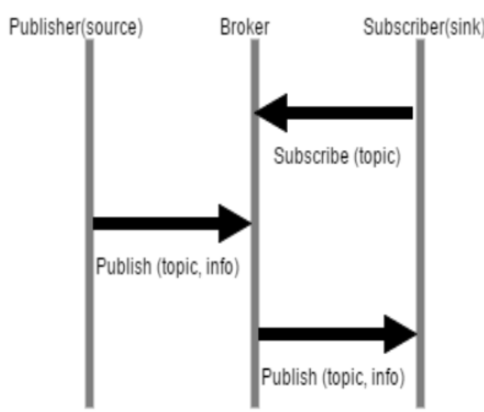


Figure 2.3: Publish/Subscribe Process Utilized by MQTT [25].

In this work, IoT technology will be used to send greenhouse information, such as the status of sensors and actuators, via the MQTT protocol. Furthermore, it will also be configured to receive information published by the MQTT client server, such as setpoint definitions and remote commands for the greenhouse.

Chapter 3

Methodology

This chapter will present the methodology used to achieve the objectives of the study, specifying the functional and non-functional requirements, the technologies used, and the general architecture of the system.

3.1 System Requirements

The system requirements were implemented in order to achieve the specific work objectives established in Section 1.3. These requirements are separated into Functional Requirements (FR) and Non-Functional Requirements (NFR).

There is a consensus on the definition of functional requirements, which can be described as the specific functionalities that the system must be able to perform, that is, what the system must do. On the other hand, the approach to non-functional requirements varies between authors and project types. Non-functional requirements encompass aspects such as performance, usability, reliability, and security, among others, which are not directly related to the system's functionalities but are equally essential to ensure its effectiveness and acceptance [26].

Therefore, in this work, functional requirements outline the actions and architecture of the system, while non-functional requirements establish the context and limits within which the system must operate. In other words, functional requirements define the "what" and "how" of the system, while non-functional requirements define "how well" the system must perform those functions.

3.1.1 Functional Requirements

System functional requirements were divided into physical requirements and system architecture requirements. The physical requirements are related to the construction of the greenhouse prototype, while the architectural requirements cover the control algorithm requirements in addition to the communication architecture requirements.

Functional Requirements of the Physical System

- FR01 - The prototype must have an area designated for planting microgreens.
- FR02 - The prototype must be closed and transparent to guarantee isolation between the internal and external environment, with openings only for ventilation.
- FR03 - The prototype must have an area designed to drain excess water.
- FR04 - The prototype must include a water reservoir for irrigation.
- FR05 - The prototype must have adequate insulation to protect the electronic circuit.
- FR06 - The prototype must have adequate dimensions for growing microgreens.
- FR07 - The device must be capable of acquiring and monitoring data on temperature, air humidity, soil humidity, and lighting from the greenhouse's internal environment through sensors.

Functional Requirements of the Architecture System

- FR08 - The system must publish the acquired data at 10 seconds intervals.

- FR09 - The device must be capable of receiving manual commands to manipulate the actuators through the MQTT protocol.

- FR10 - The device must be capable of receiving reference instructions for specific crops, detailing how to control various parameters such as temperature, air humidity, soil moisture, and illuminance. These instructions should be differentiated for day and night periods.

- FR11 - The device must have non-volatile memory to avoid loss of information during reboots.

- FR12 - The device must contain a local Human-Machine Interface (HMI) for viewing information and controlling actions in real-time.

- FR13 - The device must have an algorithm capable of performing control actions on each variable in the system based on the data acquired and the instructions received.

- FR14 - The system must distinguish between day and night periods for its control action.

3.1.2 Non-Functional Requirements

Just like functional requirements, non-functional requirements are also divided between the physical part of the system and the architecture and control algorithm part.

Non-Functional Requirements of the Physical System

- NFR01 - The prototype must include a transparent lid or cover.

- NFR02 - The system must be scalable to support the addition of new sensors and actuators in the future.

- NFR03 - The system's local HMI must be intuitive and easy to understand the information displayed.

- NFR04 - The system must operate efficiently, minimizing energy consumption.
- NFR05 - System maintenance must be simple, allowing software updates without significant interruptions.
- NFR06 - The system must be robust and resistant to failures, ensuring high availability and reliability.
- NFR07 - The accuracy of temperature, air humidity, soil humidity, and lighting sensors must be within an acceptable margin of error, according to the technical specifications of the sensors used.
- NFR08 - The system must be modular, allowing it to handle multiple cultivations simultaneously.

Non-Functional Requirements of the Architecture System

- NFR09 - The system must be able to connect to the Internet.
- NFR10 - The system must provide automatic reconnections with Wireless Fidelity (Wi-Fi) and the MQTT broker in case of connection losses.
- NFR11 - The communication protocol between the device and the monitoring system must be suitable for IoT applications.
- NFR12 - The structure of the data model used to send and receive information must be standardized.
- NFR13 - The system must have backup and data recovery mechanisms to avoid loss of information during restarts or failures.

3.2 System Architecture

To better understand the information flow and IoT application of the system, this section will explore the design of the system architecture. Figure 3.1 shows a graphical representation of the system architecture, which has a structure divided into layers, covering all levels of the information process, from acquisition by sensors to monitoring by local and remote interfaces.

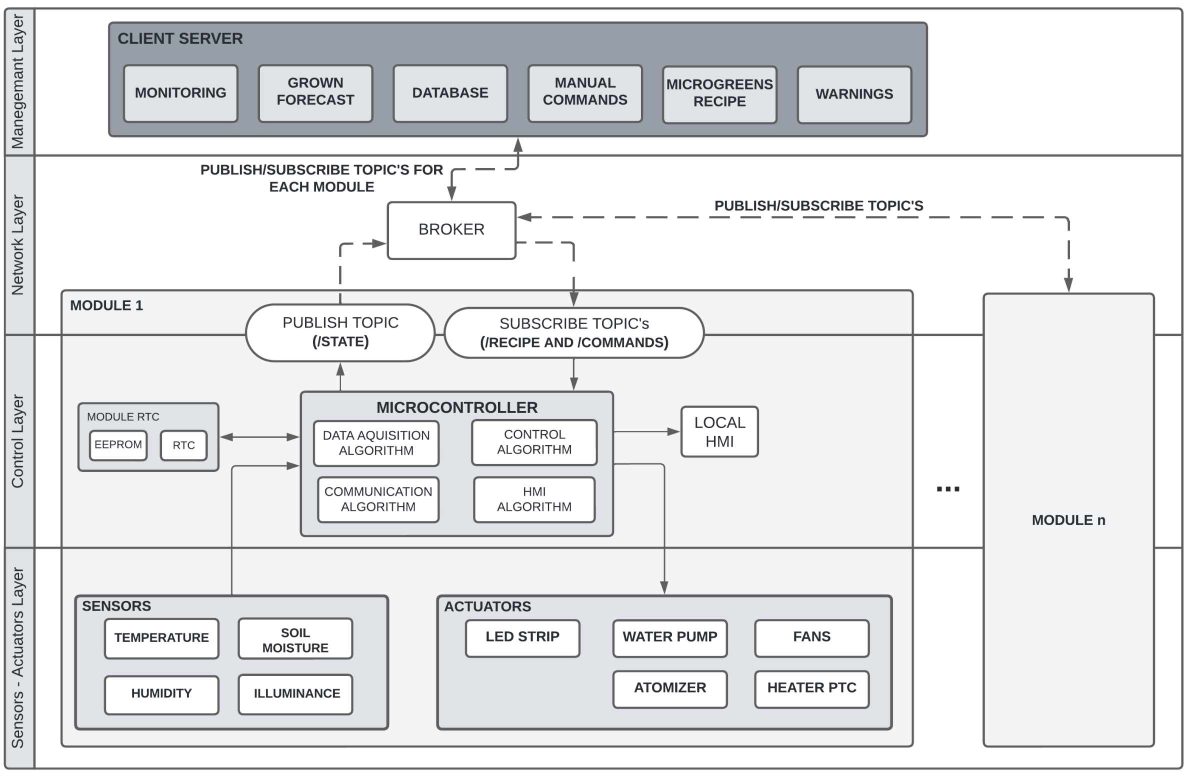


Figure 3.1: System Architecture.

3.2.1 Sensors and Actuators Layer

The bottom layer refers to the physical environment, where sensors and actuators operate in the greenhouse. The sensors are responsible for acquiring data inside the greenhouse and sending this data to the microcontroller. Variables such as temperature, air humidity, soil humidity and light are considered in this work to be monitored. Actuators manipulate controlled environmental variables. The system has an LED strip to control lighting, a water pump for irrigation, fans for ventilation, a Positive Temperature Coefficient (PTC) heater for heating the greenhouse, and an atomizer for air humidification.

The sensors and actuators used in this system will be detailed throughout the document and must comply with the requirements proposed in Section 3.1.

3.2.2 Control Layer

This layer refers to the control layer, which incorporates several functional blocks deployed in a microcontroller responsible for handling data acquisition, executing the control of the variables in the greenhouse, managing the communication with the cloud layer, and supporting the HMI. It is essential to highlight that the communication between the microcontroller, sensors, actuators, and HMI is carried out through physical connections using wired cabling, and the communication between the microcontroller and the MQTT broker is done via a Wi-Fi connection.

In addition, there is a Real-Time Clock (RTC) module associated with the microcontroller, whose function is to provide the current date and time to the microcontroller and have an internal non-volatile memory to store important system data.

Details about the microcontroller configuration and the developed algorithm will be explored throughout Chapter 4.

3.2.3 Network Layer

The Network layer refers to the mechanisms that allow the control system to communicate with the external monitoring system, located in the cloud. In this work, this communication is carried out using the MQTT protocol, since it is widely recognized in the IoT context due to its lightness, scalability, and ability to be used in embedded systems, among other characteristics. This protocol follows the publish/subscribe model, where a broker manages the exchange of information, sending data from the published device to the devices that are subscribed [27].

Considering the practicality and reliability of the MQTT protocol for IoT solutions, it was chosen as a fundamental part of this project. As illustrated in Figure 3.1, the communication broker provided by MQTT is responsible for managing all message exchanges through publications and subscriptions. Additional details about the MQTT broker used and specific topics will be explored throughout Chapter 4.

3.2.4 Management Layer

The Management layer is responsible for allowing a client system to perform functions such as monitoring, crop growth predictions, sending manual commands to the greenhouse, generating alerts, and defining variable setpoints through recipes (the concept of recipes will be better explained in Section 4.3.5) and storing data in a database.

In this project, a complete management system was not developed, only the communication structure was prepared for integration with this client system. Additional details about the management system and its functionalities will be covered in the next chapter.

Chapter 4

Development

This chapter addresses the development of the system, covering everything from the physical structure of the greenhouse, the electronic hardware devices to the control algorithm. This development follows the requirements presented in Section 3.1.

4.1 Greenhouse Structure

This section presents the construction of the greenhouse structure, including the preparation of the initial system design and its final construction.

4.1.1 Initial Design

For the initial development of the greenhouse structure, the TinkerCAD platform was used, which allows Three-Dimensional (3D) modeling in an intuitive and accessible way [28]. This platform was essential to create the three-dimensional model of the structure developed for this project.

As Figure 4.1 illustrates, the proposed physical greenhouse comprises cultivation modules and a water reservoir. The concept in relation to the modules would be the freedom to have as many modules as the user desires, all of which would be integrated into the reservoir's power supply and irrigation system.

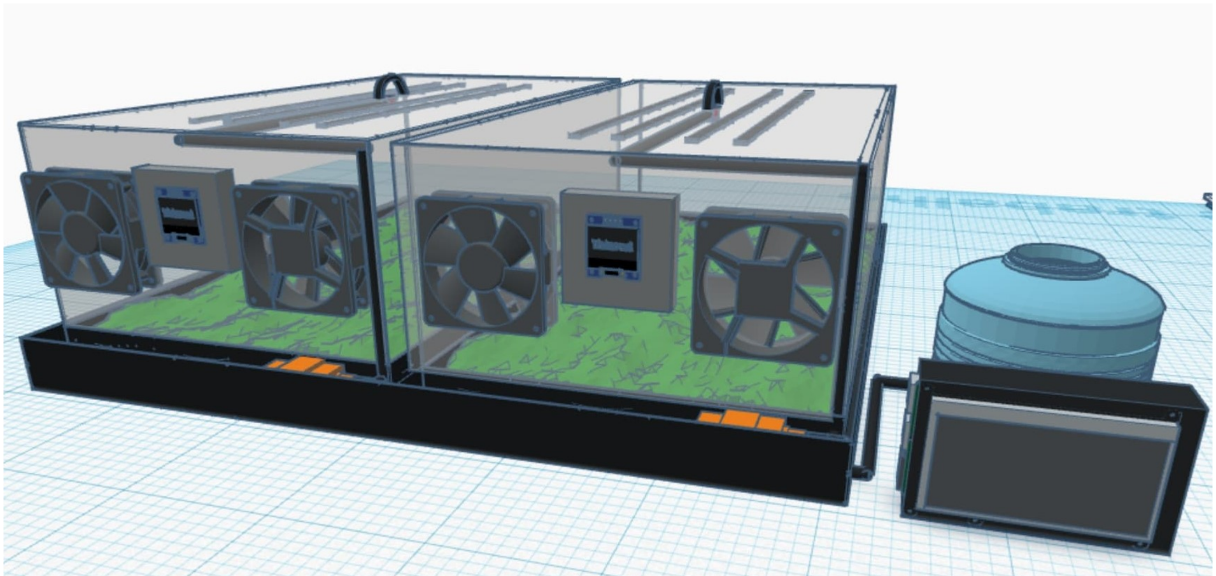


Figure 4.1: Initial Design and 3D Modeling of the Prototype - Overall View.

Each module would be independent and responsible for its own cultivation, and each one could have a different culture of microgreens. The approach of one module next to another is just a layout concept; however, thinking about a system with multiple modules, one could imagine a vertical structure where each compartment of this structure would include a different module.

Regarding dimensions, the initial prototype would have, for each module, a lid measuring 28 cm wide, 56 cm long, and 15 cm high, made of a transparent material. The perforated tray of each module would have similar dimensions to the lid, being 28 cm wide and 56 cm long. At the junction of two parallel modules, as represented in Figure 4.1, the largest base tray that would support both would be 60 cm wide, 60 cm long, and 4 cm high. These dimensions were adopted due to commercial models of trays for small plantations.

A small water reservoir was also modeled on the side where the water for irrigation of the system comes from. The dimensions of the reservoir were arbitrary, fulfilling only the role of physical representation. The interconnections between the reservoir and the modules were made using 16 mm^2 diameter pipes, commonly used in garden irrigation.

In front of the reservoir, there is a box with a screen, representing a Raspberry Pi board with its own display, just exemplifying a monitoring system.

On the front of the stove module, there is an attached box that contains the electronic components, such as the microcontroller and the connections for the system's sensors and actuators. There is also the design of a front display, corresponding to the HMI interface, where it is possible to check the current information of the stove, such as, for example, the last value read by all sensors and the current status of the actuators. More details about the information presented on the display will be provided throughout the document.

Internally, it is possible to observe in more detail the LED strips for lighting, the piping, the sprinkler for irrigation and the fans for cooling, as illustrated in Figure 4.2.

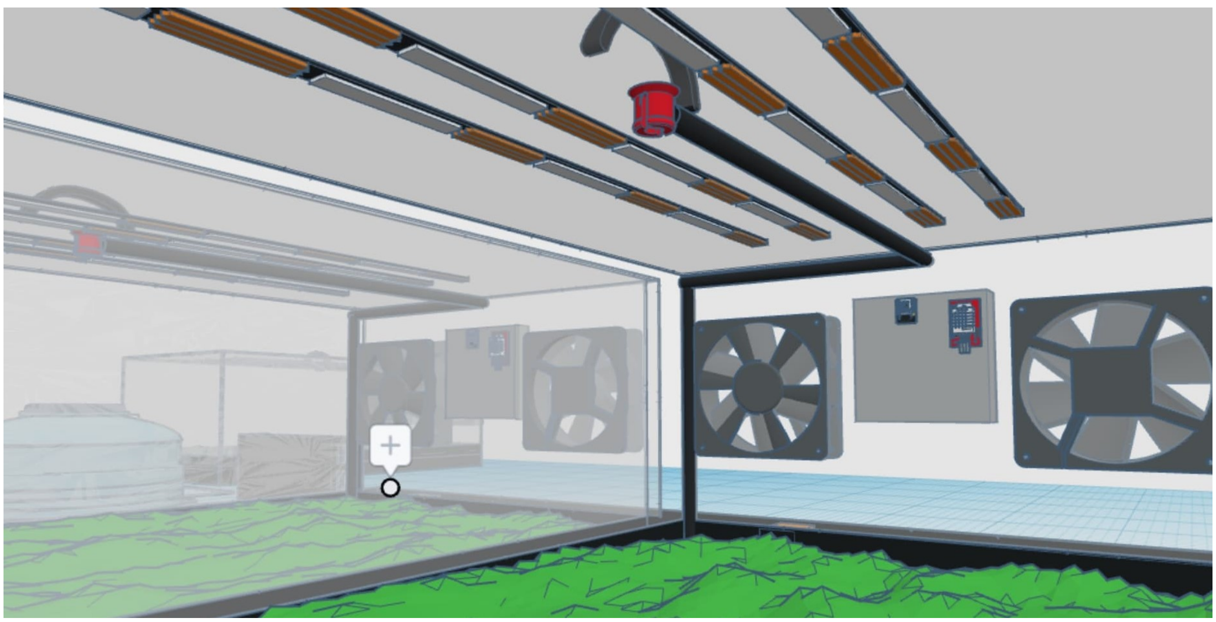


Figure 4.2: Initial Design and 3D Modeling of the Prototype - Inside View.

The fans were installed in pairs for each module. In this configuration, one fan is responsible for injecting air into the greenhouse, while the other is responsible for extracting air, ensuring efficient air circulation.

LED strips were placed on the top cover of the greenhouse for a more uniform lighting distribution inside. Likewise, the use of a central sprinkler for crop irrigation was designed.

It is important to highlight that this was the initial conception of the project, which went through several changes throughout development, which will be discussed in more detail in the following sections.

4.1.2 Physical Prototype

Following inspiration from the 3D model of the greenhouse, the physical structure of the system was developed. Some design changes were inevitable due to practical challenges. One of these modifications was the construction of just one module as a prototype, instead of two, as suggested in the initial design. This change occurred because it would be more interesting to build just one module initially and, after checking the strengths and weaknesses of this prototype, implement a second, more functional and robust module. However, there was no time to build the second prototype. Therefore, this project is limited to constructing and implementing a single greenhouse module as a prototype.

Figure 4.3 shows the system fully built and operational.

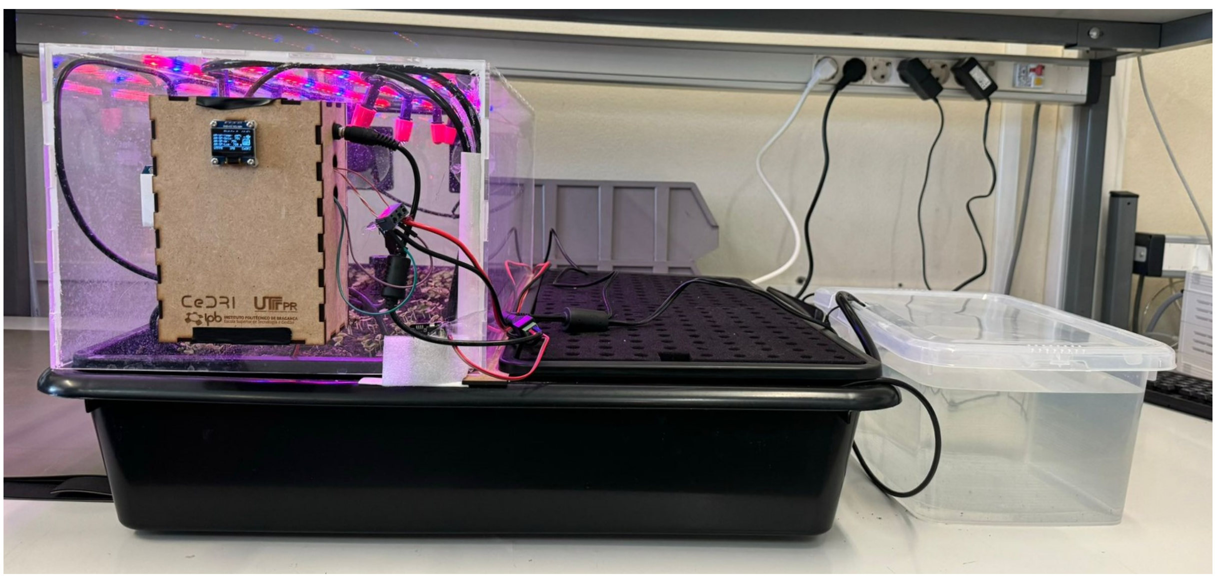


Figure 4.3: Physical Prototype.

Lid and Box Modeling

The modeling of the prototype cover and circuit box was carried out using the MakerCase online software in conjunction with SolidWorks. In MakerCase, the lid and box were sketched with their dimensions and finger-type joints in the connections. In SolidWorks, the necessary holes were made in each plate. Both the modeling and the cutting of the plates were carried out with the help of the Instituto Politécnico de Bragança (IPB) mechanics laboratory (FABLAB).

The material used to manufacture the transparent lid was acrylic, chosen for its characteristics of resistance, transparency and malleability for laser cutting. The width and length dimensions remained the same as the initial design, 28 cm and 56 cm, respectively, only the height was changed from 15 cm to 21 cm, as the need for this addition was observed when modeling the lid so that the system would not be too low.

Using an acrylic plate measuring 50 cm wide, 125 cm long, and 3 mm thick, all the necessary cuts were made to assemble the transparent lid. To glue the plates, a generic glue and sealant was used.

Another change in relation to the original design of the stove is the position of the fans, which are now located at the rear of the stove and no longer at the front, along with the circuit box. This change was necessary after verifying that there was not enough space to accommodate the two fans and the box on the same side, as the circuit box ended up having larger dimensions than expected.

To assemble the circuit box, Medium Density Fiberboard (MDF) material was used, due to its low cost and opacity, which helps to hide and isolate the electronic circuit internally.

In this circuit box, some cuts were made on the sides and front. The front cut was made to fix the display and improve the visualization of the interface. The cut on the left side facilitates access to the programming cable for the microcontroller used. On the right side, there are holes to fit the system power supply. To glue the plates, instant glue was used.

To recognize the institutions collaborating in this project, their logos were inserted on the front of the box.

Water Reservoir

The reservoir used in the project, illustrated in Figure 4.4, has a capacity of 8 liters. It was necessary to open a small hole in the top cover to allow the piping to pass through. The 16 mm^2 tubing was retained due to commercial conditions. Inside the reservoir there is a submerged water pump, connected to the tube responsible for irrigating the greenhouse.

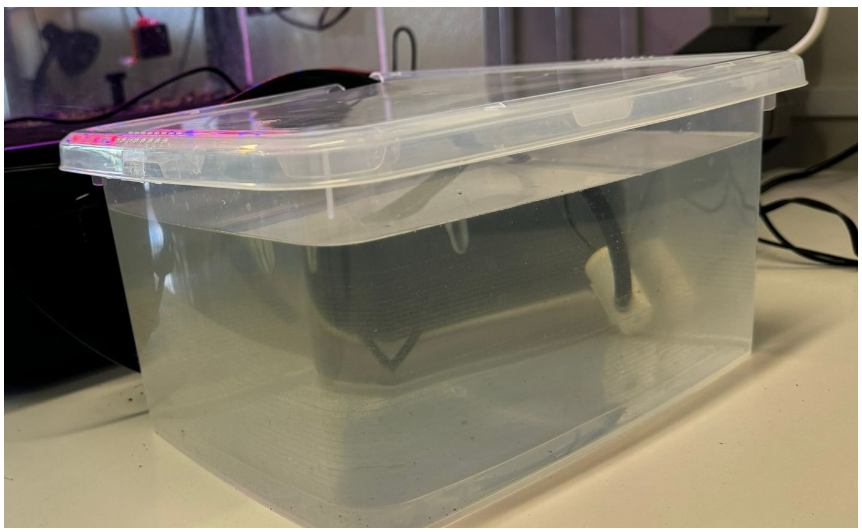


Figure 4.4: Water Reservoir.

Cultivation Tray

The main tray where the microgreens will be planted is 28 cm wide, 56 cm long and 3 cm high. This tray has scattered holes so that excess water can drain. Figure 4.5 illustrates the tray used in more detail.

Figure 4.6 represents the larger tray with dimensions of 60 cm wide, 60 cm long and 12 cm high, differing only in height compared to the initial design, which was only 4 cm. This change was necessary due to the difficulty in finding commercial trays with the projected dimensions. This tray is also used to store water drained.

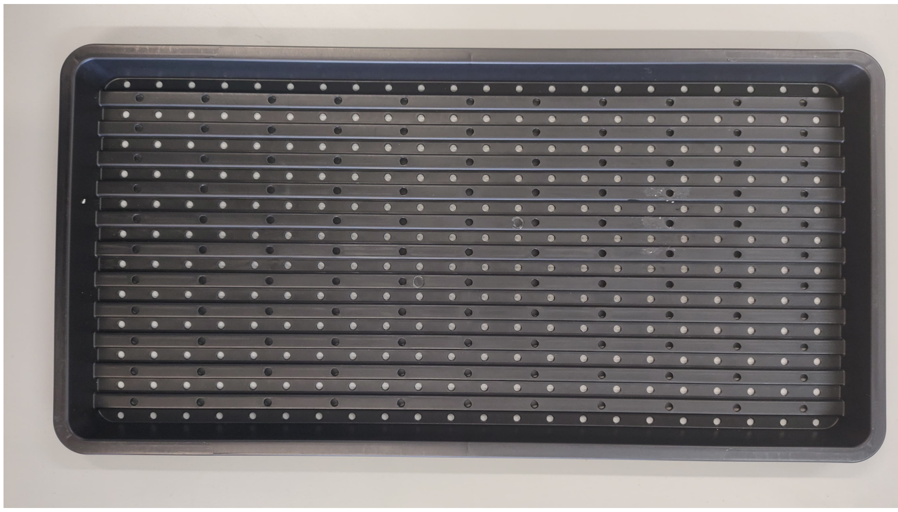


Figure 4.5: Perforated Tray.

Figure 4.6: Base Tray.

4.2 Hardware Device

After an overview of the greenhouse prototype, this section will explore in detail the hardware used in this project.

4.2.1 Microcontroller

The microcontroller used in this project was the ESP32 DevKit v4. This development board was chosen due to its particular technical characteristics. The ESP32 DevKit v4 has a large number of inputs and outputs, which makes it ideal for connecting various sensors and actuators. In addition, this board includes communication protocols such as Inter-Integrated Circuit (I^2C), Serial Peripheral Interface (SPI) and Universal Synchronous Asynchronous Receiver Transmitter (USART), providing a wide variety of options for interacting with external devices. One of the main advantages of ESP32 DevKit v4 is the presence of an integrated Wi-Fi module, which significantly simplifies network connectivity and eliminates the need to use extra modules [29].

Figure 4.7 shows a representation of the used ESP32.

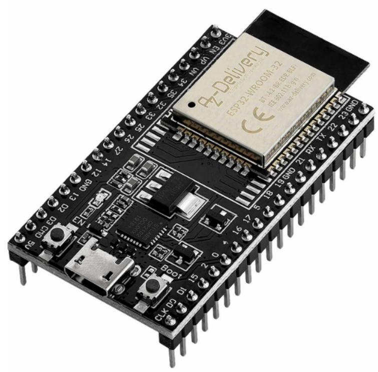


Figure 4.7: Microcontroller ESP32 DevKit v4.

4.2.2 Power Supply

The power source used in this project, illustrated in Figure 4.8, is 12V and supports a maximum power of 250W, well above what is needed for a single module. However, in a scenario with several modules, the use of a single source for the system becomes interesting, thus justifying the choice of this source.

To make possible to power the microcontroller with this source, a voltage regulator was used, specifically the 7805, which receives a voltage of 12V at its input and delivers 5V at its output, as shown in Figure 4.9.

The ESP32 microcontroller operates with voltages of 5V or 3.3V. In this project, the board was powered with 5V, but the microcontroller's digital inputs and outputs operate at voltages of 3.3V. Therefore, the sensors used were selected to operate at this voltage

Figure 4.8: Power Supply 12V.

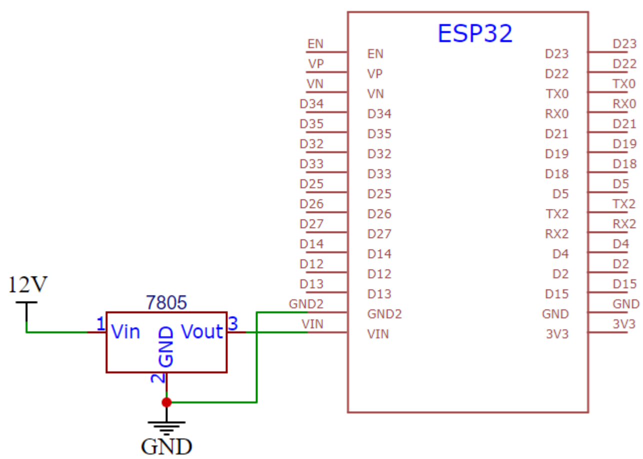


Figure 4.9: Input Voltage Circuit.

level. Furthermore, actuators, as will be discussed in the 4.14 section, typically operate at 12V, requiring dedicated drive circuits for each of them.

The 12V voltage was chosen as the standard for this project due to the need for many actuators to require at least this voltage. Another approach would be to use alternating voltage for some actuators, such as heaters, but a lower voltage was initially chosen for safety reasons and to avoid mixing low voltages, such as 5V and 12V, with 220V on the same circuit board.

4.2.3 Real-Time Clock

For this project, a RTC was used to perform checks and record times. Furthermore, the use of an RTC that had Erasable Programmable Read-Only Memory (EEPROM) was preferred to allow the storage of important system information, such as the revenue received by the management system, in addition to storing the current operating modes, among other data.

With these requirements established, the RTC DS3231 was chosen, as illustrated in Figure 4.10. This device is capable of storing 32KB of information in memory, far beyond what is needed for the project, as well as providing a highly reliable and accurate time base. Its communication protocol is the I^2C.

Figure 4.10: Real-Time Clock DS3231.

4.2.4 Human-Machine Interface

For the HMI, a mini Organic Light-Emitting Diode (OLED) display of 128x64 pixels was used, illustrated in Figure 4.11, which allows the insertion of a lot of information, especially when compared to a Liquid Crystal Display (LCD) for electronic projects. The display used uses I^2C as its communication protocol, being arranged on the same bus as the RTC in the physical prototype.

Six operating screens were developed for the HMI display. Each screen was designed to show important information about the greenhouse, so that the operator can see exactly

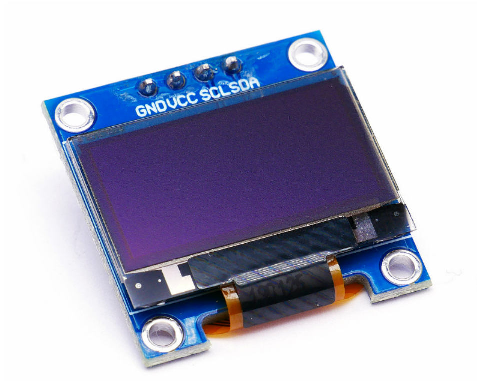


Figure 4.11: Display OLED 128x64.

how the module is configured and what it is currently controlling. Figure 4.12 shows an example of these screens.

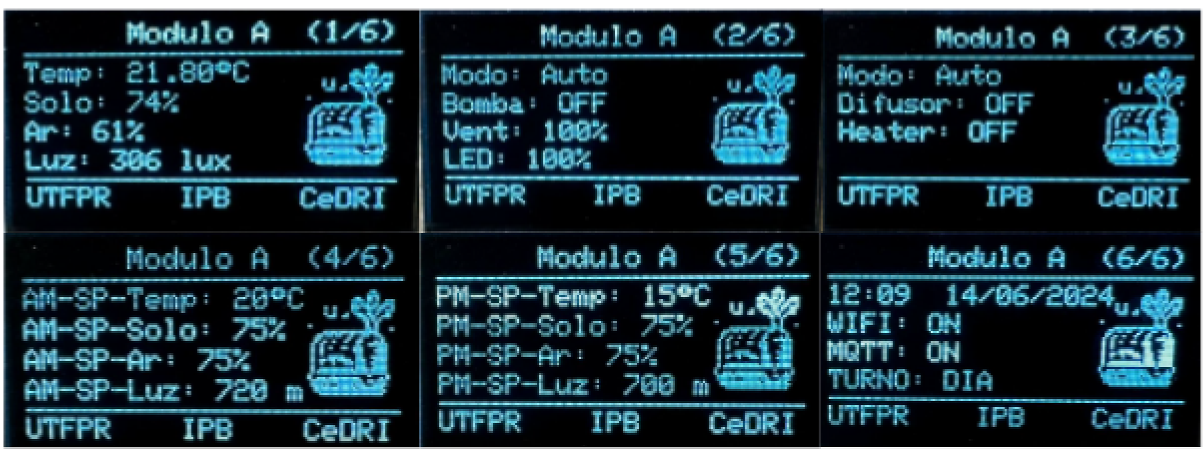


Figure 4.12: Screens HMI Interface.

The first screen of the interface shows the last value read by the sensors, indicating the temperature, air and soil humidity values, in addition to the current illuminance of the greenhouse. The second and third screens show the states of the actuators, indicating whether they are activated or not, in addition to informing whether the stove is in automatic or manual mode. The fourth and fifth screens show the current recipe setpoints, indicating the reference value for temperature, air and soil humidity, in addition to the lighting time using the LED strip. The fourth screen shows the values for the daytime period and the fifth screen for the nighttime period. Finally, the sixth screen shows general greenhouse information, such as the current date and time, shift indication (day or night), as well as Wi-Fi and MQTT status, indicating whether the communication is working correctly or if there are problems.

Additionally, on all screens it is possible to see which screen is currently being displayed, from the first to the last, the indication of which module is being shown, such as A, B, C, the acronyms of the supervising entities of the project, and the project logo, which consists of a mini greenhouse.

4.2.5 Sensors

The sensors are responsible for acquiring the signals of the system variables, which are, for this project, temperature, air humidity, soil moisture and illuminance, all from the point of view of the greenhouse interior. Figure 4.13 shows a representation of the sensors used in this project.

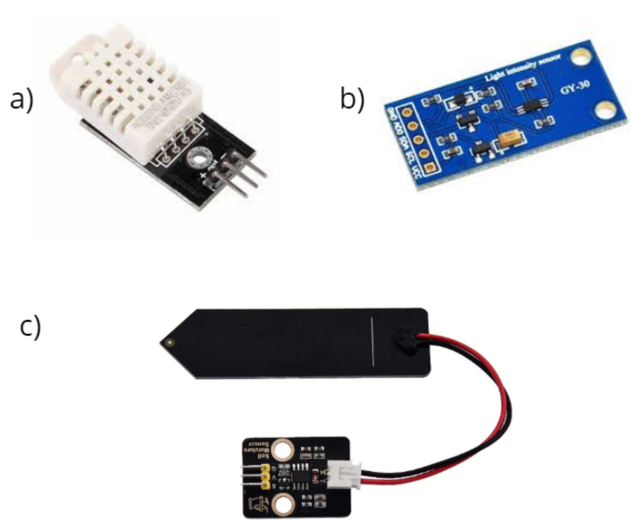


Figure 4.13: Sensors. a) DHT22 - Temperature and Humidity b) BH1750 - Illuminance c) KS0510 - Soil Moisture.

Temperature and Humidity Sensor

The air temperature and humidity sensor used is the DHT22, as illustrated in Figure 4.13. This sensor was developed by Aosong Electronics Co., Ltd. [30]. The operating range of this sensor is from -40°C to 80°C for temperature and from 0% to 100% for relative air humidity. It has an accuracy of ± 0.5°C for temperature and ± 2% for relative humidity and an operating voltage of 3.3V to 5V

This sensor was chosen for its technical characteristics combined with a low cost, in addition to being able, through an integrated module, to measure both the temperature and humidity of the environment, optimizing the project itself.

The sensor was fixed to the internal side of the greenhouse, so that it was close to the circuit box, facilitating its connection with the microcontroller.

Soil Moisture Sensor

The soil moisture capacitive sensor used is the KS0510 from Keyestudio [31], as illustrated in Figure 4.13. The measurement range of this sensor varies from 0 to 100%, providing an accurate indication of the soil moisture level.

The sensor was inserted directly into the soil inside the greenhouse, close to the plants, to monitor soil moisture. Based on this information, the irrigation control system will be able to act and maintain soil moisture within acceptable levels. Therefore, this sensor and the monitoring of these variables are of great importance for this project.

This sensor was chosen because it is capacitive, as there are also resistive sensors that have lower quality and are very sensitive to corrosion. Furthermore, this specific sensor has its circuit board separated from the measurement region, preventing the circuit from coming into contact with excessive humidity and increasing its useful life.

Illuminance Sensor

The brightness sensor used is the BH1750 built into the GY-30 board, as illustrated in Figure 4.13. This sensor was developed by Rohm Semiconductor [32] and is available as a module from Keyestudio [33]. The measuring range of this sensor is 0 to 65535 lux, with an accuracy of ± 4 lux. The sensor was fixed to the internal side of the greenhouse opposite the DHT22 so that the sensor could capture the ambient light correctly.

The importance of this sensor is due to the fact that it allows monitoring of the lighting that the microgreen crop is receiving. In principle, in this project, this variable is not being used directly in the lighting control system, serving only as an indication of ambient lighting.

This sensor is connected to the microcontroller using the I^{2}C protocol.

4.2.6 Actuators

This section presents the actuators used to control the stove, which are illustrated in Figure 4.14. It is important to highlight that there were some modifications to the originally planned actuators, in addition to the inclusion of new ones that were not foreseen in the initial scope of the project.

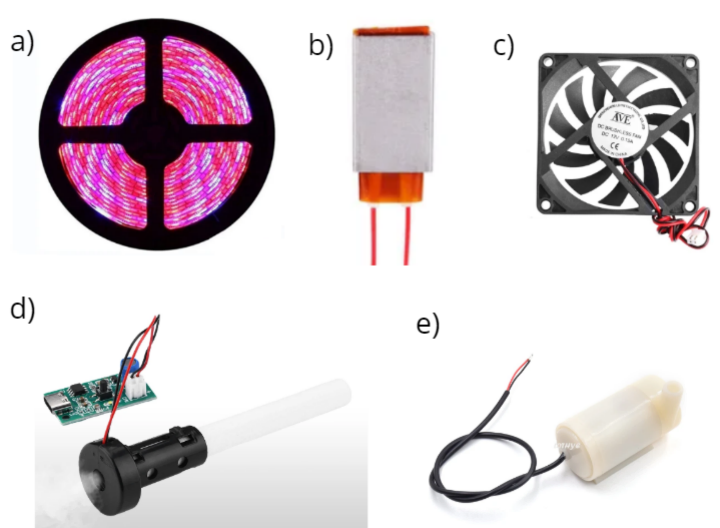


Figure 4.14: Actuators. a) LED Strip Grow b) Heater PTC c) Ventilators d) Atomizer e) Water Pump.

LED Strip Grow

Artificial lighting of greenhouses with LEDs has proven to be an efficient alternative, especially during periods of low natural light. The use of red and blue LEDs has effects directly related to plant growth and quality. While red light is more effective in producing biomass and elongating leaves, blue light significantly increases the levels of flavonoids, bioactive compounds important for health, in several plant species [34].

For this project, an LED strip containing red and blue colors was used in a 5:1 ratio, that is, for every five red LEDs, a blue LED was used, as illustrated in Figure 4.15. The tape has IP67 waterproof protection specifications and operates at a voltage of 12V. The tape used in the greenhouse is approximately 1.5 meters long. With these specifications, the tape is suitable for use as lighting inside the greenhouse, ensuring the maintenance of an environment with high humidity and having a desired voltage level, compatible with other actuators present in the system.

Figure 4.15: LED Strip in the Prototype.

A drive circuit was developed for this actuator, consisting of an optocoupler and a transistor in series. The optocoupler, used in this project to isolate the digital electronic part from the power part, has its input connected to the corresponding output of the microcontroller. Then, the output of the optocoupler is connected to the base of the transistor, as shown in Figure 4.16. This configuration allows a complete variation of the LED's light intensity levels, which can be changed between levels of 0% and 100% using a digital output of the Pulse Width Modulation (PWM) type on the microcontroller.

It is important to highlight that the CNY17-3 optocoupler model used in this project is the one and the transistors used follow the BD241 model. These models were chosen because their voltages and operating powers are compatible with the circuits designed.

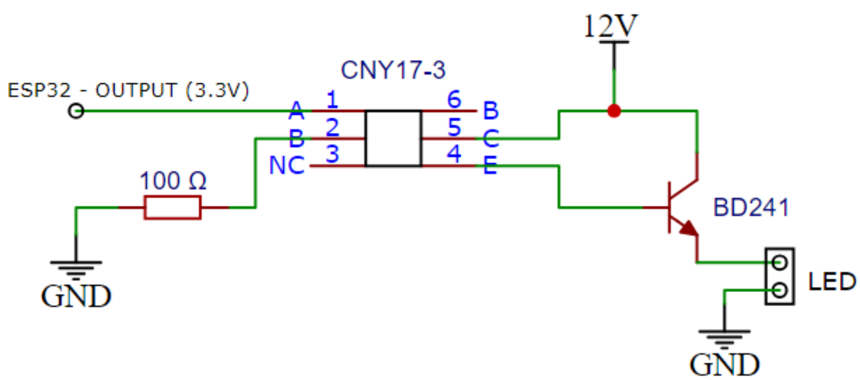


Figure 4.16: LED Strip Drive Circuit.

With the use of the LED strip for lighting, aligned with the project specifications, and the implementation of a dedicated circuit for its activation, it is safe to say that this actuator was developed and applied appropriately. Its correct integration contributes significantly to achieving the objectives and meeting the requirements established by the project.

Heater PTC

For this project, it was necessary to include a PTC heater, even though it was not initially included in the scope, due to the need to more accurately control the room temperature, especially on colder days. This type of heater is known for its ability to regulate internal temperature through its own electrical resistance. When the temperature increases, the PTC resistance also increases, which reduces the electrical current and, consequently, the increase in temperature.

PTC heaters are widely used in applications that require precise temperature control. For this project, a PTC heater was chosen with 12V supply voltage specifications and the ability to reach a maximum temperature of 80°C, with an average power consumption of 12W. This choice guarantees not only heating efficiency but also safe temperature control within the desired parameters for the greenhouse environment.

Figure 4.14 item “b” illustrates the heater used. However, an adaptation was necessary to extract maximum performance from the heater. A heat sink coupled with a mini cooler was implemented to optimize heat transfer to the environment, using an adhesive thermal paste. This adaptation provides greater heating efficiency, ensuring precise control of the temperature inside the greenhouse.

The heatsink and cooler used are represented by Figure 4.17. The dimensions of this heatsink are 40 mm wide, 40 mm long, and 10 mm high, while the cooler has the same dimensions but with 15 mm high and a 12V power supply, with a power consumption of 1.2W.

Figure 4.17: Heat Sink for Heater.

It is also important to mention that it was decided to use two heaters due to the dimensions of the greenhouse. Figure 4.18 shows how the heaters were arranged in the greenhouse, being fixed to the sides of the greenhouse.

The heaters used were connected in parallel, so that both are activated simultaneously. To control these heaters, a drive circuit based on the use of a relay was developed. This circuit includes three main components: an optocoupler, a transistor and a relay. Similar

Figure 4.18: Heaters in the Prototype.

to the LED driving circuit, the optocoupler is connected in series with the transistor, however, in this case, the transistor is also connected in series with the relay coil, activating it when activated.

It is important to highlight that the relay used has an operating voltage of 12V and supports a maximum current of 10A. The normally open connection of the relay was used to connect the heaters. Figure 4.19 shows in more detail the circuit developed and applied.

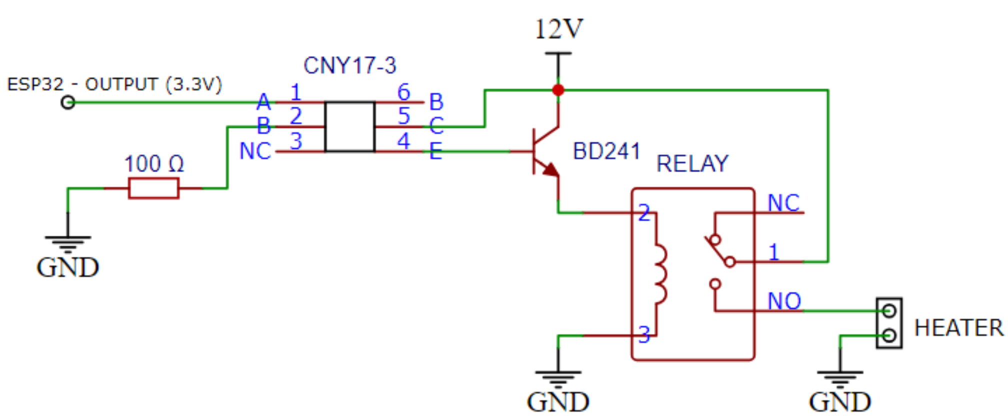


Figure 4.19: Heater Drive Circuit.

Ventilator

Ventilation in a greenhouse is essential, whether forced or natural, as air exchange helps to maintain CO_2 conditions within ideal limits for plant development, in addition to helping to cool the interior [35]. As the proposal for this work is limited to an indoor greenhouse, natural ventilation is very limited.

Aiming to better control the system's cooling and air exchange, a double ventilation system was included, consisting of the use of two fans. One fan injects air inside, while the other extracts air from the inside. The fans used are 80 mm wide, 80 mm long and 20 mm thick. They operate with a voltage of 12V and a power consumption of 0.9W each.

The fans were placed at the rear of the greenhouse, unlike the initial design in which they were located at the front. This change was necessary, as mentioned at the beginning of this chapter, due to the limited space available caused by the circuit box, which had to be larger than expected.

Figure 4.20 shows in detail how the fans were positioned in the prototype. They were fixed using instant glue.

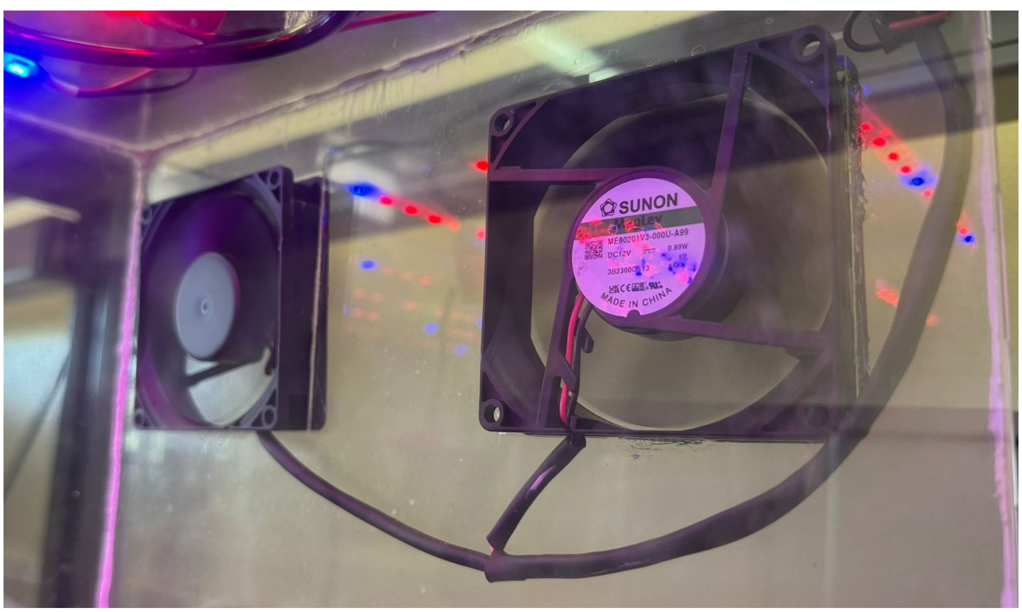


Figure 4.20: Ventilators in the Prototype.

The fan drive circuit is very similar to that of lighting, with the only difference being that for its control, the Darlington transistor configuration was chosen. This configuration consists of the use of two transistors in series, causing an increase in the overall gain. This configuration was necessary due to problems in testing in which, using just one transistor, it was not able to control the fans correctly.

Figure 4.21 shows how the fan drive circuit was designed.

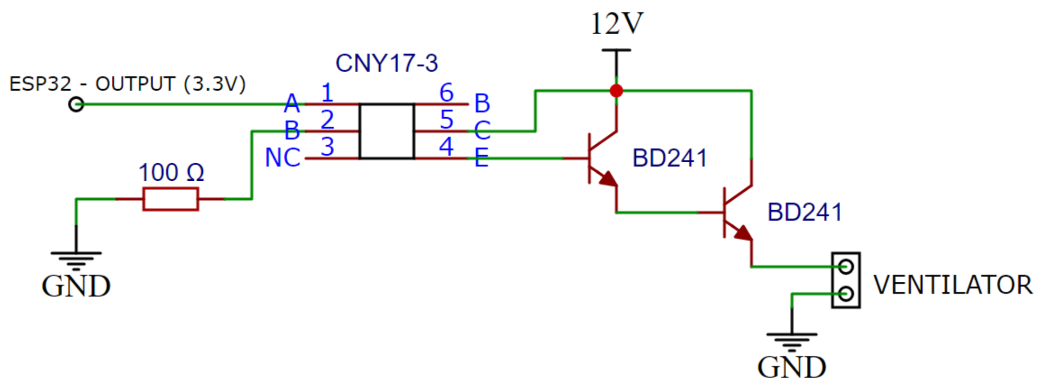


Figure 4.21: Ventilators Drive Circuit.

Atomizer

As noted in item "d" of Figure 4.14, for this project an atomizer was added to help maintain air humidity inside the greenhouse. This actuator was inserted into the system after the initial project, due to the need to be able to act directly on the humidity in the air, and not passively with just irrigation.

The atomizer used consists of a piezoelectric tablet that converts electrical energy into vibrations. When this tablet is in contact with a wet surface, in this case a cotton tube, it transfers the energy of these vibrations to the water, generating mist. Figure 4.22 shows in more detail the atomizer used, containing a piezoelectric tablet, a cotton tube, a plastic support for fixing the tablet and the tube, as well as a plate to activate the actuator.

To activate this actuator by the microcontroller, the same circuit that drives the LEDs was used, consisting only of an optocoupler in series with a transistor. The output of this transistor is connected to a 5V connection point on the atomizer board, allowing

Figure 4.22: Atomizer Details.

the microcontroller's digital output to activate the atomizer correctly, as illustrated by Figure 4.23.

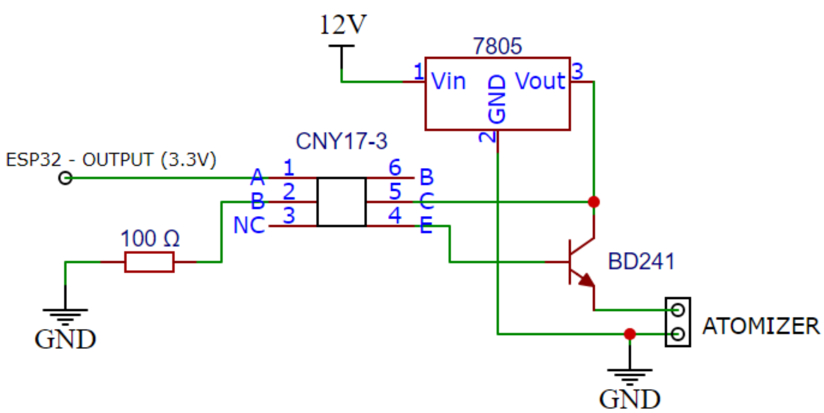


Figure 4.23: Atomizer Drive Circuit.

Furthermore, this actuator does not have intermediate levels of activation, being only considered a digital component, that is, it is on or off.

Figure 4.24 shows how the atomizer was inserted into the greenhouse's internal environment. It was preferred to insert the component in the center of the greenhouse due to a better distribution of the mist generated by it. It is important to highlight that a piece of foam was used to better fix the actuator in the center of the stove, in addition to improving water absorption from the environment. The cotton cane uses soil moisture to moisten itself.

Figure 4.24: Atomizer in the Prototype.

Water Pump

Irrigation is one of the most crucial elements for creating an environment favorable to the germination and growth of plants, especially microgreens. As noted in item “e” of Figure 4.14, a submersible water pump was used in this project. This water pump in question works with an operating voltage between 3V and 6V, with an average consumption of 1.2W of power.

To connect the irrigation circuit, 16 mm^2 and 4 mm^2 hoses were used, as well as some connectors to make the connection between these tubes possible. Figure 4.25 shows the elements used to form the project’s irrigation circuit.

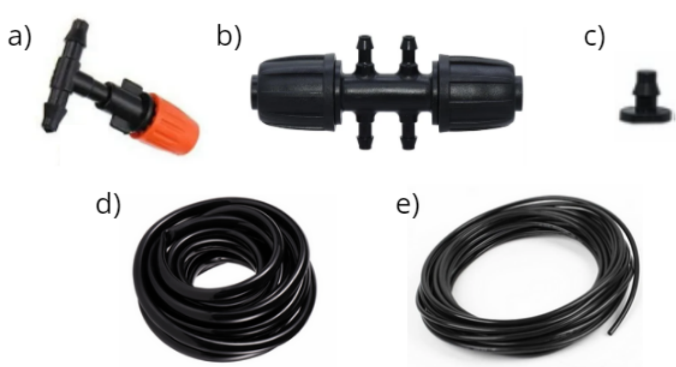


Figure 4.25: Irrigation Elements.

Item “a” is the mist nozzle, already equipped with its 4 mm^2 hose adapter. He is responsible for spraying the water, ensuring its uniform distribution inside the greenhouse. Item “b” is the adapter between the 16 mm^2 and 4 mm^2 hoses. Item “c” is a terminal to close the excess points of the 4 mm^2 hose. Item “d” is the 16 mm^2 hose, used in the irrigation circuit between the water pump and the modules. Finally, item “e” refers to the 4 mm^2 hose, used inside each module to connect the mist nozzles.

Figure 4.26 shows the interconnection between the 16 mm^2 hoses and the 4 mm^2 hoses, using the adapter. It is important to highlight that the internal irrigation circuit in the module is closed, that is, it begins and ends in this same adapter, in order to maintain constant pressure throughout the circuit, minimizing interference caused by loss of charge in each mist nozzle.

Another important factor in using this adapter is that it is ready to allow the connection of another module in series. In Figure 4.26, it is possible to see that there is another 16 mm^2 inlet, although it is currently closed with a hose with a plug, as this project operates with only one module. However, if it were necessary to add another module, simply connect the hose to this additional input.

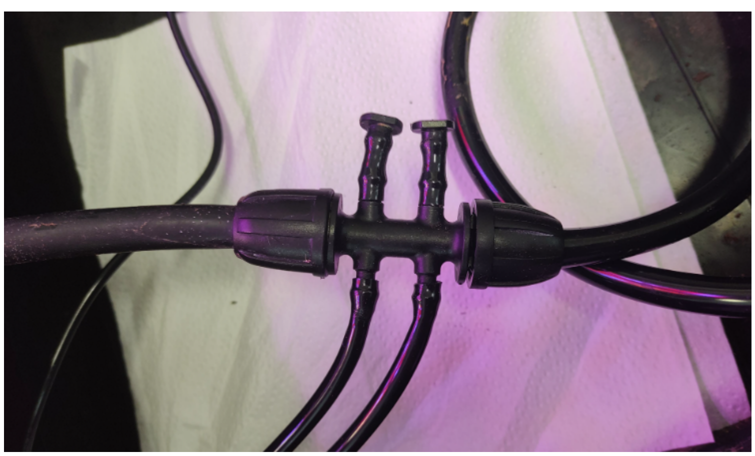


Figure 4.26: Hose Adapter Connection.

Figure 4.27 illustrates how the irrigation system was arranged inside the modules, with two parallel rows containing four mist nozzles each. This configuration was developed based on tests carried out, aiming to achieve a uniform distribution of water in all corners of the greenhouse.

Figure 4.27: Irrigation System in the Prototype.

For the drive circuit, the same base circuit was used as the other actuators, consisting of an optocoupler and a transistor to turn on the water pump. The difference lies in the supply voltage, as the water pump operates in a range of 3V to 6V, while the rest of the system operates at 12V. As a solution, a voltage regulator was inserted, specifically the 7805 component, which reduces the continuous voltage from 12V to 5V. The connection diagram is illustrated in Figure 4.28.

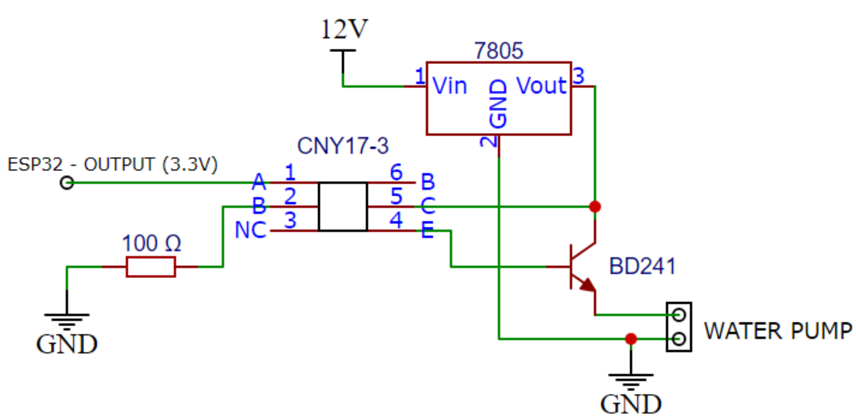


Figure 4.28: Irrigation Driver System.

4.3 Control System Programming

In this section, the development of the control algorithm is discussed, together with the communication solution and the HMI. Aspects such as communication with sensors, update routines, waiting times and the integration of all these elements operating on the ESP32 microcontroller will be covered.

It is important to highlight that the system used to program the microcontroller was PlatformIO, integrated with Visual Studio Code. PlatformIO is an open source development platform that supports a wide range of embedded systems and frameworks. One of its biggest benefits is the ease of integration with various community libraries. In turn, Visual Studio Code is a highly configurable code editor, widely adopted for its intuitive interface and extensibility.

For this project, four libraries were developed to organize common processes, each applied to different purposes in the algorithm as a whole. The libraries developed include WiFi and MQTT communication, general plant control, data acquisition and processing by sensors, as well as a general-purpose library for the HMI and functions that did not fit into any other specific library.

The algorithm developed together with the libraries and other files necessary for the complete execution of this project are available at the link in the Appendix A.

4.3.1 Main Code: Setup and Loop Functions

In the main code of the project, there is, in sequence, the import of the necessary libraries, definitions of global variables, creation of class objects present in the libraries, prototype of the main functions of the algorithm and, finally, the *setup* and *loop*.

In the *setup* function, the initial configuration occurs and is performed once when the microcontroller is turned on. In this function, the General Purpose Inputs/Outputs (GPIOs) are defined, which are configured as inputs or outputs. Furthermore, the initialization functions of the HMI display, the function to initiate WiFi and MQTT communication, indicate the topics to be subscribed by MQTT, initiate data acquisition by sensors, and activate the algorithm control functions are invoked. Additionally, during system startup, setpoint values are retrieved from the EEPROM memory present in the RTC, ensuring that, in the event of system restart, all setpoint variables are loaded correctly.

Among the global variables used, there are four that are timing definitions for calling specific functions of the algorithm. The first of these is `CONTROL_INTERVAL`, which determines the frequency of data acquisition by the sensors and the calculation of the control action, essentially determining the system's sampling rate.

The second is `COMMS_INTERVAL`, which defines the communication quality check interval. If there is a problem, a reconnection routine, defined in the developed communication library, will be executed.

The third is `PUB_INTERVAL`, which defines the interval between publications of greenhouse data, both via MQTT and the serial interface.

Finally, we have `SCREEN_INTERVAL`, which defines the time interval for changing the HMI interface screens automatically and on a timed basis.

For each of these variables, specific times were defined and will be detailed in Chapter 5.

Basically, the *loop* function performs temporal checks and analysis of the conditions of each of the previously mentioned variables. In other words, it controls and manages the calls to the functions that operate the entire algorithm.

4.3.2 Communication Library

This library was developed with the purpose of meeting WiFi and MQTT communication requirements, managing network connections and data to be published or subscribed to.

It is based on a class called `Communication`. In the main code, an object of this class is created with the following parameters: name of the WiFi network to which you want to connect, password for that network, MQTT broker address, defined MQTT port, MQTT user and password.

In addition to the parameters, there are some methods that have been developed, such as *begin*, which initializes WiFi and MQTT communication, indicating whether there was success or failure.

There is also the *loop* method, which performs a routine check to analyze the quality of the connection. If there are problems, reconnection attempts are made to stabilize the connection.

Another important method is *subsTopic*, which subscribes to the topic provided as an argument, indicating to the broker which topics will be subscribed to.

There is also the *publish_mqtt* method, which takes as an argument a string of data formatted in JavaScript Object Notation (JSON) for publication via MQTT, along with the name of the topic where the information will be published.

Finally, there is the *callback* method, which is called whenever there is a subscription to be made.

In this way, following the functions developed and applied, it was possible to meet the communication requirements for this project, with the tests carried out in Chapter 5 being better addressed.

4.3.3 Sensors Library

The sensing and data acquisition library was created with the purpose of meeting the sensor reading requirements and carrying out the necessary conversions. In addition to the sensors, this library is also responsible for providing the current time reading, in communication with the RTC DS3231 module.

This library is also class and object based, the class of this library is called `Sensors`. In the main code, an object of this class is created, whose parameters include a vector to store the values of the acquired sensors and the GPIO pins used for the DHT (air temperature and humidity) sensor and the soil moisture sensor.

As methods of this class, we initially have *begin*, which initializes the sensors, focusing on communication with the DHT and on I^2C communication with the brightness sensor and the RTC module for time acquisition, informing through communication serial if there was any initialization problem.

To acquire data from sensors, the *readSensors* method is called, which reads each sensor involved. Communication with the DHT sensor is done through its own library *DHT.h*, with temperature and humidity acquisition being done using the *readTemperature* and *readHumidity* methods of this library, respectively. For the brightness sensor, a library was also used to manage this exchange of information, *BH1750.h*, so that the current brightness value is received through the *readLightLevel* method.

Soil moisture is acquired analogically, with the value read varying between 0 and 4096, configuring a 12-bit analog-to-digital conversion reading. Therefore, the *map* function was used to convert the scale of the read value for humidity, varying between 0 and 100%. However, in order to make the conversion correctly, it was necessary to calibrate the minimum and maximum values, as the values of 0 and 4096 are the limits of the analog reading, where 0 represents 0V and 4096 represents 3.3V. The test carried out is further detailed in Chapter 5.

4.3.4 General Library and Data Structure

The library with general functions was developed in order to clean up the main code. In this library there is no developed class, just a compilation of algorithm functions.

Among them are the functions responsible for creating the HMI interface screens shown in Figure 4.12, having a total of six screens and a function FOR each of them.

Furthermore, there is the serial printing function, where all the information present on the HMI interface is also shown on the serial interface.

There are also communication functions with the EEPROM memory, including a memory initialization function, through the I^2C protocol, another function for reading the values saved in the memory, such as the setpoints of the current recipe, and finally, the save information in that memory. This last function is called whenever there is a change in revenue from the monitoring system.

In addition to the functions presented, there is also the function of unifying greenhouse information for publishing data using the JSON structure.

Data Structure

Aiming to standardize data models for the future, especially in the context of the IoT, the organizations FIWARE Foundation, TM Forum, IUDX and OASC joined forces in a collaborative initiative. This collaboration aims to support the adoption of a reference architecture and compatible common data models that support a digital marketplace of interoperable and replicable smart solutions across multiple sectors, starting with Smart Cities. This initiative, called Smart Data Models, seeks to create a solid foundation for the interoperability and replicability of technological solutions. Smart data models include four main elements: the technical schema, the specification in a human-readable document, a URI with a working URL containing basic information about the attributes or entities, and example payloads for the NGSIv2 and NGSI-LD versions. All data models are public and free, guaranteeing users three basic rights through the [36] licensing mode.

For this work, the data model from the Weather section of Smart Data Models, available at the link in Appendix A, was used as standard. This model contains information about the identification of the greenhouse, location of the device, as well as providing data such as temperature, air and soil humidity, ambient lighting, among others.

However, the original data model only contained sensor instances, without including actuator values, which are essential for this project. Therefore, this data model was augmented with more specific information, such as the status of all actuators, information regarding the current shift of operation, and whether the operation mode is manual or automatic. Just like the standard model, the modified model specified for this project is also available at the Appendix A link.

4.3.5 Control Library

The control library is one of the most essential in this project, as it contains the logic to perform the control action of the actuators. The control action on each actuator was briefly explained in Section 4.14, and is further detailed in this section.

Recipe Definition

Before going into more detail about the library, it is necessary to clarify further how the recipes are defined.

Cultivation recipes are setpoint values that must be followed, divided into two shifts: day and night. For each shift, there are reference values for temperature, air humidity, soil humidity, lighting and light exposure time.

The temperature setpoint defines the greenhouse temperature reference value, and the objective is to remain close to this value.

Soil humidity and air humidity are seen as minimum values to be followed, where the value defined as setpoint for these variables must be equal to or greater than them. In this project, it was decided to always remain close to the established reference value.

Regarding lighting, there are two parameters: the first is the intensity value of the LED strip, which varies from 0 to 100%, indicating the intensity of the light. The second parameter refers to the exposure time, that is, how many minutes the LED strip will remain on during that shift.

The day and night shifts each comprise 12 hours of the day. Defined internally in the microcontroller logic, daytime mode begins at 7 am and ends at 7 pm, when the night period begins.

Therefore, between the periods indicated above, there is a control action that varies between day and night, in order to meet the project requirements, aiming to better condition the plant's growth.

Library's Functions

For this library, a `Controls` class is created. Similar to the sensor library, this class has as parameters a vector with the current states of the actuators, another vector for the states of the environment variables, a vector containing the current setpoints to be followed, the operating mode (automatic/manual), a vector with the control action of each actuator and, finally, two vectors: one for the current time and the other as an auxiliary buffer for the time.

As a class method there is *begin*, which initializes the actuators with a null value. And *generalControl* that performs the control action on all actuators.

Control Action

The control action is defined within the *generalControl* method, where the greenhouse's operating condition is initially analyzed, determining whether it is in automatic or manual mode, as configured by the user in the monitoring system. In manual mode, all commands sent by the monitoring system to the microcontroller are executed. On the other hand, in automatic mode, these commands are ignored. Figure 4.29 represents this process.

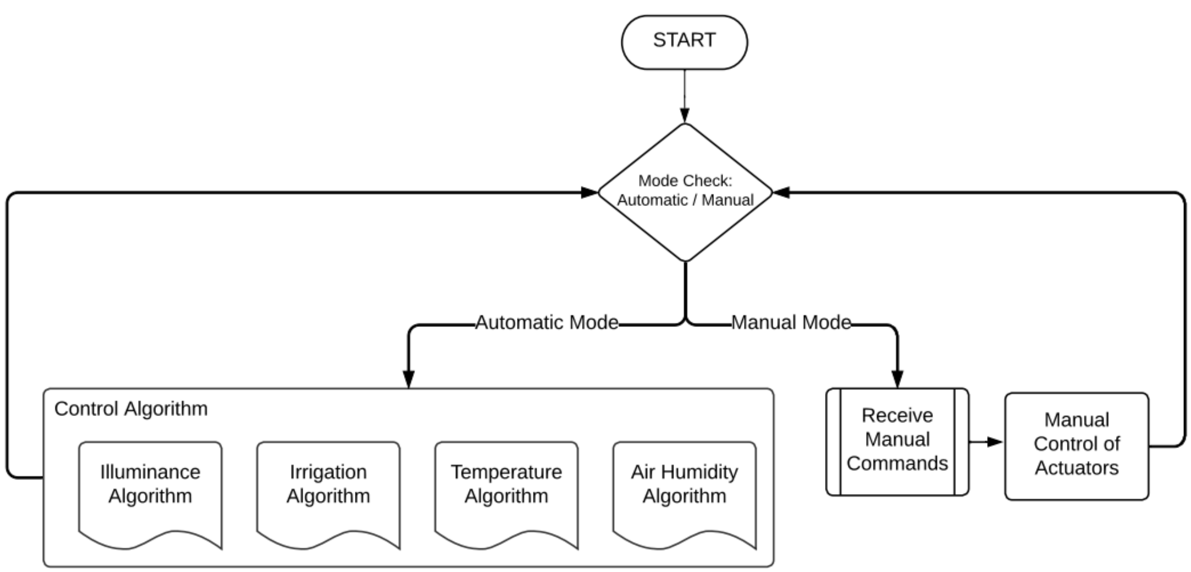


Figure 4.29: Control Flowchart.

During automatic mode, the system operates autonomously. Starting with lighting control, a calculation is carried out based on the start time of the shift (7am for daytime and 7pm for nighttime), adding to this time, the value of the light exposure time defined by the recipe, determining the moment of switching off artificial lighting from LED strips. Thus, the system keeps the lighting on at the intensity defined by the shift setpoint until the scheduled shutdown time. When this time is reached, the lighting is turned off. Each shift is limited to a maximum of 720 minutes, corresponding to 12 hours of operation. Figure 4.30 represents this algorithm.

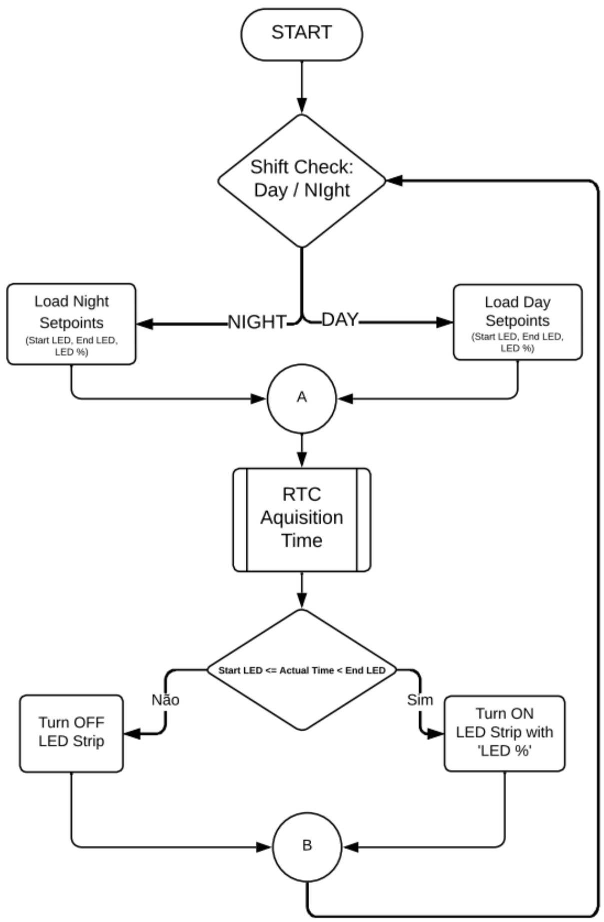


Figure 4.30: Illuminance Control Flowchart.

For the irrigation system, the control strategy includes three main parameters: *timeOnIrrig*, which specifies how long the water pump should remain active when activated; *deadTimeIrrig*, which defines the minimum interval between successive pump activations;

and *timeWaitIrrig*, which indicates how long soil moisture must remain below the reference value before the pump can be started again.

In short, whenever soil moisture falls below the setpoint defined by the recipe, a timer is started (defined by *timeWaitIrrig*). After this timer ends, the water pump is activated for the time specified in *timeOnIrrig*. Subsequently, a new check is performed after the time interval defined by *deadTimeIrrig*.

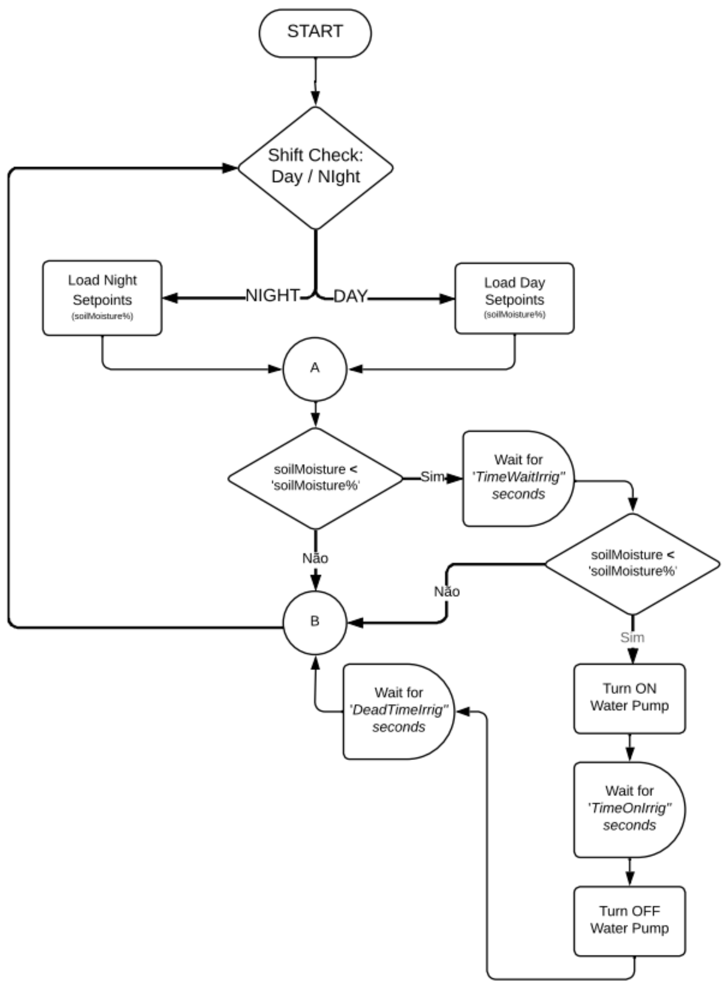


Figure 4.31: Irrigation Control Flowchart.

This irrigation control system is designed to be robust against incorrect soil moisture readings, ensuring plants are not over-watered. The significant time between the activation of the water pump and the increase in soil moisture, observed in the tests, justifies the use of the *deadTimeIrrig* parameter, for example. Figure 4.31 represents this control strategy.

To control air humidity, a hysteresis approach was adopted. It was observed that the greenhouse system tends to increase air humidity over time, due to the closed environment and humid substrate. Thus, the atomizer's trigger point is defined by the air humidity reference value minus an additional value (defined by the *histHumiAir* variable), while shutdown occurs when the air humidity reaches exactly the recipe reference value. It was not necessary to adopt an additional value for turning off the atomizer due to the greenhouse's tendency to maintain high levels of air humidity.

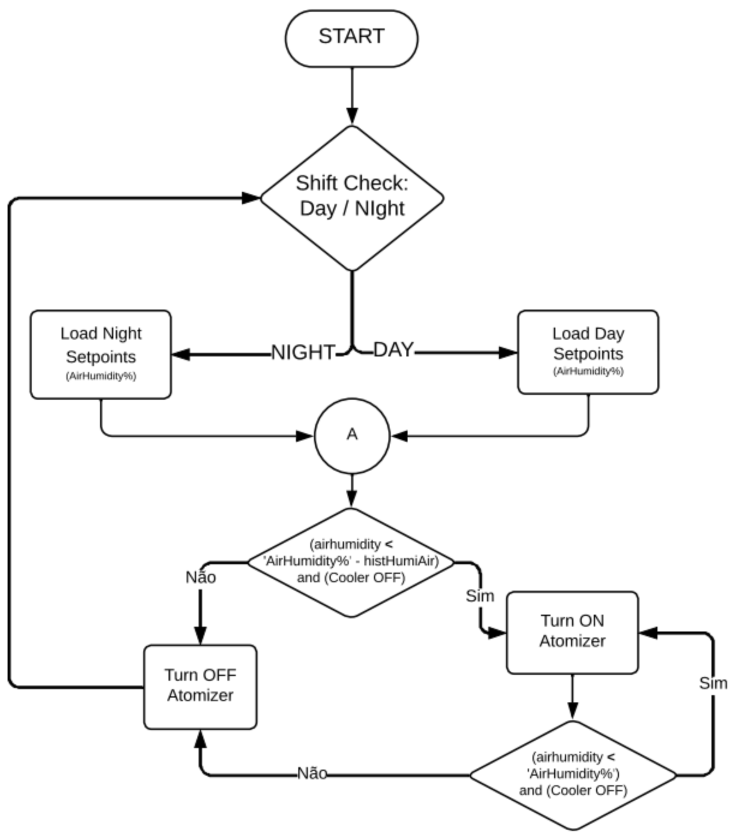


Figure 4.32: Air Humidity Control Flowchart.

Another relevant aspect to be considered is that, when the fans are activated to control the temperature, the humidity in the air tends to decrease, as explored in Section 5.1.2. This is because the air inside the greenhouse is more humid than the outside environment, and the air exchange caused by the fans reduces the humidity in the air. In this specific scenario, it has been established that if the fans are on, the atomizer cannot be activated, thus avoiding unnecessary operation. Figure 4.32 represents this control approach.

Finally, temperature control in the greenhouse is crucial, as this variable directly influences plant growth, from germination to complete development. For this purpose, two actuators were used, the heater, responsible for increasing the temperature, and the fans, which reduce the temperature by ventilating the room.

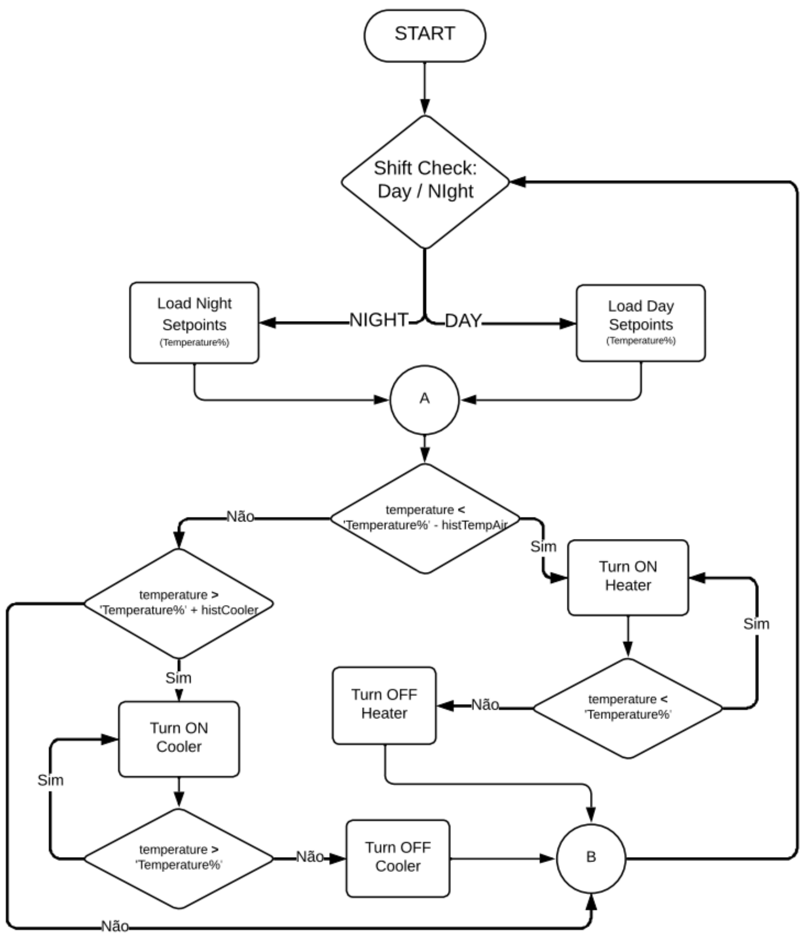


Figure 4.33: Temperature Control Flowchart.

Similar to air humidity control, greenhouse temperature control uses the hysteresis strategy to activate the heater and fans. The heater is activated when the room temperature drops below the reference value defined by the recipe, subtracting the additional hysteresis value (defined by the variable *histTempAir*), and is turned off when the temperature reaches or exceeds the reference value. In this case, it was not necessary to adopt an additional value for turning off the heater due to the time elapsed between deactivation and recognition by the temperature sensor.

To control the fans, the principle is reversed: they are activated when the temperature reaches the reference value plus the hysteresis value defined by the *histCooler* variable and are turned off when the temperature returns to the reference value. Figure 4.33 represents this temperature control strategy.

The implemented control system does not use more complex technologies, such as fuzzy logic, adaptive, predictive control, or even the classic PID, for two main reasons. Firstly, the simplicity of the applied hysteresis system offers an effective and easy-to-implement solution that meets the requirements of this project, especially by dealing with multiple internal greenhouse variables simultaneously. Secondly, in temperature control, the use of a relay to activate the heaters prevents the use of more sophisticated techniques such as PWM, necessary for advanced control algorithms such as PID or even fuzzy logic in decision making.

The effectiveness of this control system will be further explored in the Chapter 5.

Chapter 5

Tests and Results

In this chapter, the tests and results obtained with the developed prototype will be discussed, divided into No-load Control Tests and Cultivation Test with microgreens culture.

5.1 No-load Control Tests

The No-load Tests were designed with the aim of verifying the influence of each actuator on environmental variables. To achieve this, a test routine was carried out in which the actuators were activated one at a time manually, while the environmental variables were monitored. In this way, it was possible to measure the influence of each actuator on each environmental variable.

5.1.1 Heater Test

In this section, the greenhouse heating test will be discussed using the heaters used and mentioned in the previous sections. The test was carried out on May 17, 2024, between the times of 10:55 and 13:20. During this period, the heaters were turned on to the maximum and environmental variables were monitored, enabling insights into the influence of this actuator on the system to be obtained.

Figure 5.1 shows the graphs obtained of the relationship between the greenhouse's internal temperature and air humidity in relation to the heater's performance. Firstly, on the temperature graph, it is possible to clearly observe a considerable increase in temperature when the heater is turned on, tending to stabilize close to the value of 27.4 °C. The initial temperature value in the tests was the same as the ambient temperature outside the greenhouse, recorded as 17.3 °C. Therefore, it is possible to conclude that the connected heater managed to increase the greenhouse temperature by a variation of 10.1 °C in an approximate interval of 55 minutes. Another insight that is possible to obtain from this graph is the steep curve of reduction in temperature when turning off the heater, taking approximately 1 hour for the temperature to normalize.

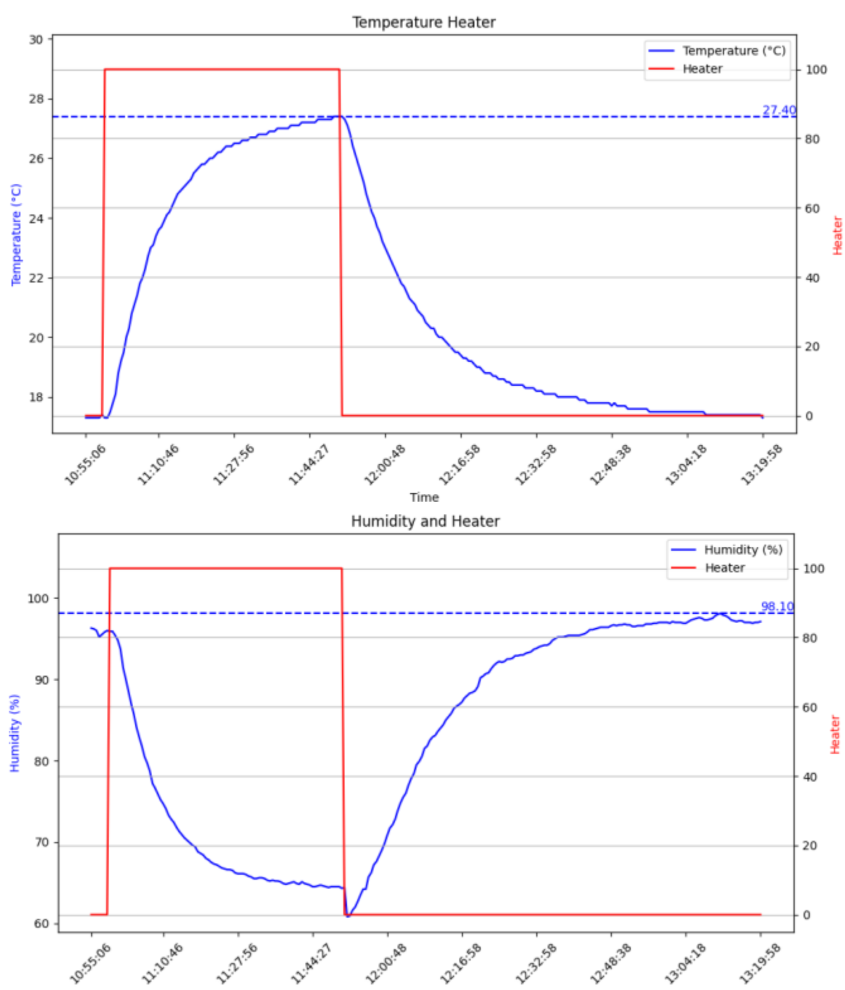


Figure 5.1: Heater Influence - Temperature and Air Humidity.

In the air humidity graph as a function of the heater, it is possible to see a drastic decrease in humidity while the heater remains on, stabilizing at around 65% while the heater is on. This highlights a direct relationship between air humidity and the heater. The tendency of the greenhouse system to be very humid is also noticeable, as when the heater is turned off, there is a significant increase in humidity, reaching approximately 98% relative humidity in 1 hour.

Figure 5.2 shows the relationship between the heater and soil humidity and ambient illuminance. The first graph, that of lighting, shows that there is no direct relationship between the heater activation and the ambient lighting. It is noticeable that the ambient lighting increases over time due to the time of day, where sunlight ended up influencing the tests a little, due to the laboratory conditions at the time of the tests.

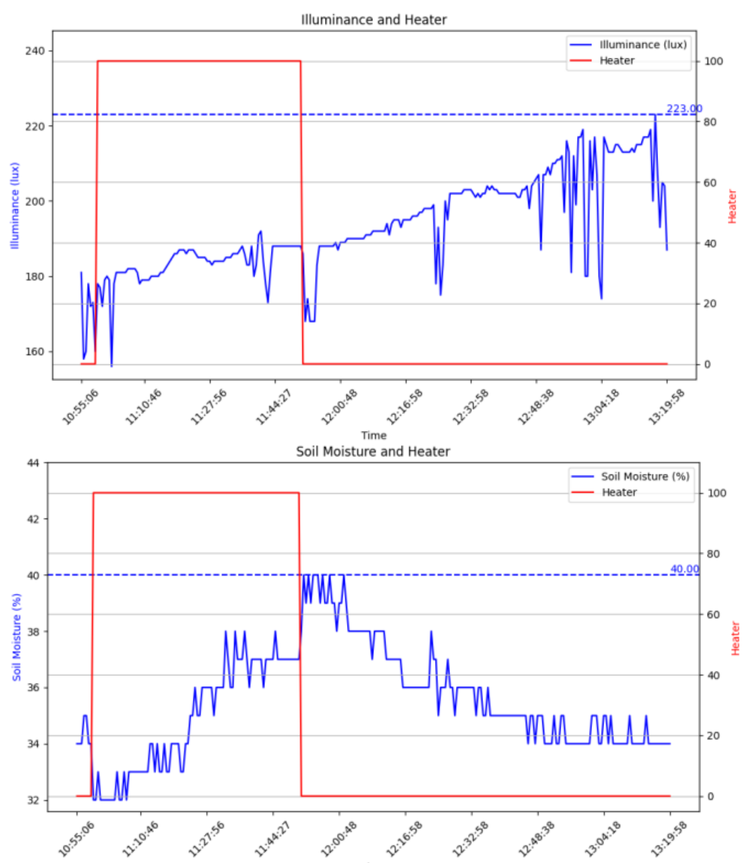


Figure 5.2: Heater Influence - Illuminance and Soil Moisture.

The second graph, of soil moisture, does not show a direct relationship, just a trend with little variation between soil moisture and the heater. As the system was not irrigated during this period, it is plausible to believe that this variation in the soil moisture variable can be interpreted as system noise.

Another test carried out for the heater was the use of automatic mode for temperature control. Figure 5.3 shows the control carried out by the heater when trying to keep the environment close to 20 °C. The hysteresis value used is 0.2 °C. In this test, it is possible to verify that, as soon as the heater turns off when it reaches 20 °C, the system temperature continues to rise up to 20.3 °C. In this way, the system cools naturally until it reaches 19.8 °C, activating the heater again. In other words, it is possible to verify that the heater operates in automatic mode with a tolerable range of 0.3 °C above and 0.2 °C below the temperature reference value.

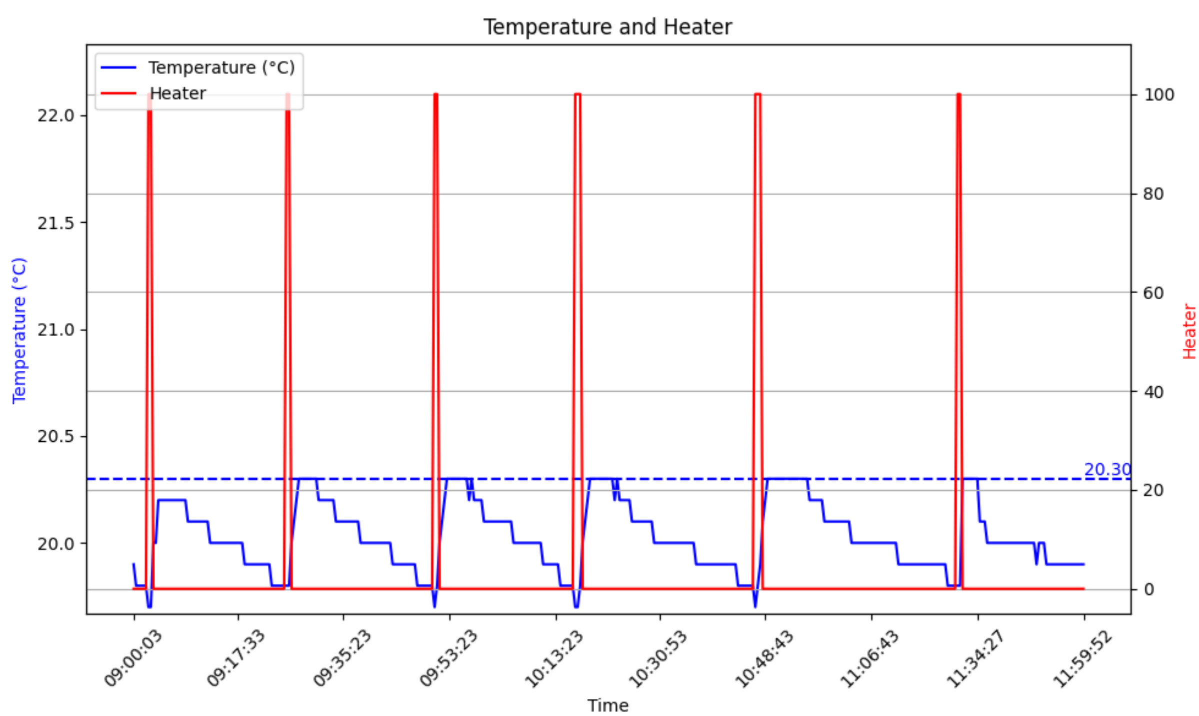


Figure 5.3: Heater Automatic Test.

5.1.2 Ventilator Test

In this section, the cooling and ventilation test promoted by fans will be discussed, and their impact on environmental variables. The test was carried out on May 17, 2024, between the times of 14:35 and 15:17. Initially, the greenhouse temperature, which was approximately 17.2 °C, was raised to 26 °C using the heater. Subsequently, the heater was turned off and the fans were turned on, and the variables were then monitored to verify the effectiveness of ventilation in cooling the space, comparing it with the natural dissipation of heat into the environment, as observed in the previous test with the heater.

Therefore, Figure B.1 shows graphs of the variation in air temperature and humidity recorded during this test. In the first graph, it is possible to observe the relationship between temperature variation and fan activation. After heating the room to 26 °C, ventilation was activated, resulting in a drastic reduction in temperature. In about 20 minutes, the temperature dropped from 26 °C to 17.5 °C. Comparing with the natural heat loss, observed in the previous test with the heater, which took approximately 1 hour for the temperature to stabilize, it can be concluded that the ventilation forced by the fans was approximately three times faster in terms of cooling.

The second graph in Figure B.1, which shows the relationship between air humidity and fans, presents an interesting dynamic. Initially, the air humidity was approximately 98.90%. When turning on the heater to raise the temperature, there was a decrease in air humidity, an effect discussed in previous tests with the heater. Subsequently, when turning on the fans, an even more drastic reduction in air humidity is observed due to the exchange of the greenhouse's internal air with drier ambient air. In general, due to the greenhouse's natural behavior of remaining humid, the external environment tends to be drier. With ventilation on, outside air enters the greenhouse, thus reducing indoor humidity, as shown in the graph. When turning off the ventilation, the internal humidity tends to increase again.

Figure B.2 shows the relationship between the fans and the illuminance and soil humidity variables. The first graph in this figure shows the greenhouse's internal lighting and the performance of the fans. It is possible to observe an unstable lighting condition, subject to variations in external lighting, as the greenhouse is transparent, and possible noise. However, it is not possible to conclude any relationship between this variable and ventilation.

In a similar way to the second graph, there is variation in soil moisture depending on the activation of the fans. There is also a variation in soil moisture levels, making it possible to see a small drop in the reading due to ventilation activation.

5.1.3 LED Strip Test

In this section, the ambient lighting test will be discussed, showing the influence of the LED strip on the lighting and whether this actuator ends up changing or impacting any other environmental variable. This test was carried out on May 18, 2024, between the times of 16:28 and 16:45.

Figure B.3 shows the relationship between the actuator and the air temperature and humidity variables. The first graph shows the variation in temperature depending on artificial lighting. It is possible to verify that there are no major changes; However, when it remains on for about 5 minutes, the temperature begins to rise gradually, but nothing very significant.

For the second graph, which relates lighting to the air humidity variable, it is observed that air humidity remains constant, without major variations, indicating that the LED strip does not directly impact air humidity.

Figure B.4 shows the relationship between the measurement of ambient brightness and soil humidity in relation to the activation of the LED strip. The first graph shows the variation in lighting, which occurred as expected. There is an increase in lighting from approximately 170 lux to 377 lux instantly, showing how much the LED strip impacts the overall lighting of the greenhouse.

For the second graph, related to the soil moisture variable, it is evident that there is no considerable variation, therefore, a direct relationship between the variation in soil moisture and the performance of the LED strips is not observed.

5.1.4 Atomizer Test

In this section, testing the atomizer depending on environmental variables will be discussed. For this test, the set of fans was activated to reduce air humidity; as demonstrated in previous tests, there is a tendency for the greenhouse to maintain high humidity levels. Therefore, the fans were activated to lower the humidity and then the atomizer was turned on to check how long it took for the humidity to reach the established limit, comparing this time with the time needed for the humidity to reach the limit naturally. The test was carried out on May 18, 2024, between 15:43 and 16:25.

Figure B.5 shows two graphs. The first is related to the ambient temperature and the second to air humidity.

Regarding temperature, it remained constant during activation, without significant variations. Therefore, it was assumed that there is no direct relationship between temperature and the atomizer.

Regarding air humidity, it is clear that there is a considerable increase in humidity when the atomizer is activated. Initially, humidity was at approximately 60%, rising to as high as 98.4% within about 8 minutes of operation. Another important detail to note is that, in just 3 minutes after the atomizer was turned on, the air humidity rose to 90%.

Compared to the natural rise in humidity observed in the heater tests, the humidity stabilized at 98.4% after approximately 1 hour naturally, while, forcedly, it took only 8 minutes. In other words, the atomizer raised the humidity in the greenhouse about 7.5 times faster than naturally.

Figure B.6 shows the relationship between the atomizer and the illuminance and soil moisture variables.

In the first graph of this figure, it is possible to see a small variation in illuminance over time, but not necessarily linked to the activation or deactivation of the atomizer. Looking at this graph in detail, it is safe to say that the atomizer does not affect ambient lighting.

In the second graph, there is a measurement of soil moisture during the test. It is possible to observe the constancy of soil moisture over time, showing that, no matter how much the atomizer is turned on, it does not end up directly influencing soil moisture.

5.2 Cultivation Test

In this section, the practical test carried out on a microgreen crop, more specifically a basil crop, will be discussed. For this test, an automatic mode was used for autonomous production management. Furthermore, basil was chosen due to its characteristics of having a relatively low germination time compared to other microgreens, which is between 5 and 8 days.

For this test, it is expected that the microgreens will germinate and develop around 21 days after germination, in addition to efficient control throughout the developed control system. A size of around 3 cm from the plant is also expected.

5.2.1 Test Specifications

The crop cultivation test was carried out between May 21, 2024 and June 18, 2024, with all information on the greenhouse's internal variables being collected and made available at the link in the Appendix A.

The recipe defined for the production of this microgreen is shown in Table 5.1.

RECIPE		
Variables	DAY	NIGHT
Temperature (°C)	20	15
Air Humidity (%)	75	75
Soil Moisture (%)	75	75
LED Intensity (%)	100	0
LED Time (min)	720	0

Table 5.1: Recipe With the Setpoints Applied to the Cultivation Test.

The values chosen for the recipe were taken as a basis because they are the most commonly applied for the production of basil, with a temperature of 20 °C being excellent for optimal germination and growth of the plant, according to the instructions on the packaging of the seed used.

In this way, it is expected that the control system will be able to keep the internal variables close to the intended values, taking into account the limitations of the system.

5.2.2 Results

The cultivation test was completed with germination taking an average of 7 days and, after this period, another 21 days of growth, totaling 28 days of testing. Figure 5.4 shows the plantation on the first day of sowing and on the last day of the test.

Figure 5.4: Cultivation Test: Left Side Shows the First Day, and Right Side Shows the Last Day.

On the last day of testing, the size of the cultivated basil was measured. Figure 5.5 shows that it was approximately 2 cm, approaching the expected value of 3 cm.

Figure 5.6 shows the average values of the greenhouse's internal environment variables throughout the cultivation period, separated between the day and night periods. For

Figure 5.5: Cultivation Test: Measurement of Basil.

temperature, a value close to 20 °C was expected during the day and 15 °C at night. However, the result showed a value of 21.77 °C for the day and 20.73 °C for the night. This variation between the expected value and the obtained value demonstrates a limitation in controlling the temperature of the greenhouse, which is unable to cool the internal air to a temperature lower than the ambient temperature, as it does not have a refrigeration system, only ventilation. Therefore, the minimum possible temperature to be reached is limited by the ambient temperature.

For air humidity, a value close to 75% was expected for both day and night. However, it will be seen later that the fans were turned on for a long time during planting to ventilate the environment and, in this condition, as discussed in Chapter 4, the atomizer is not turned on to avoid unnecessary activation. Therefore, due to the high external temperature during the testing period, air humidity control presented this limitation.

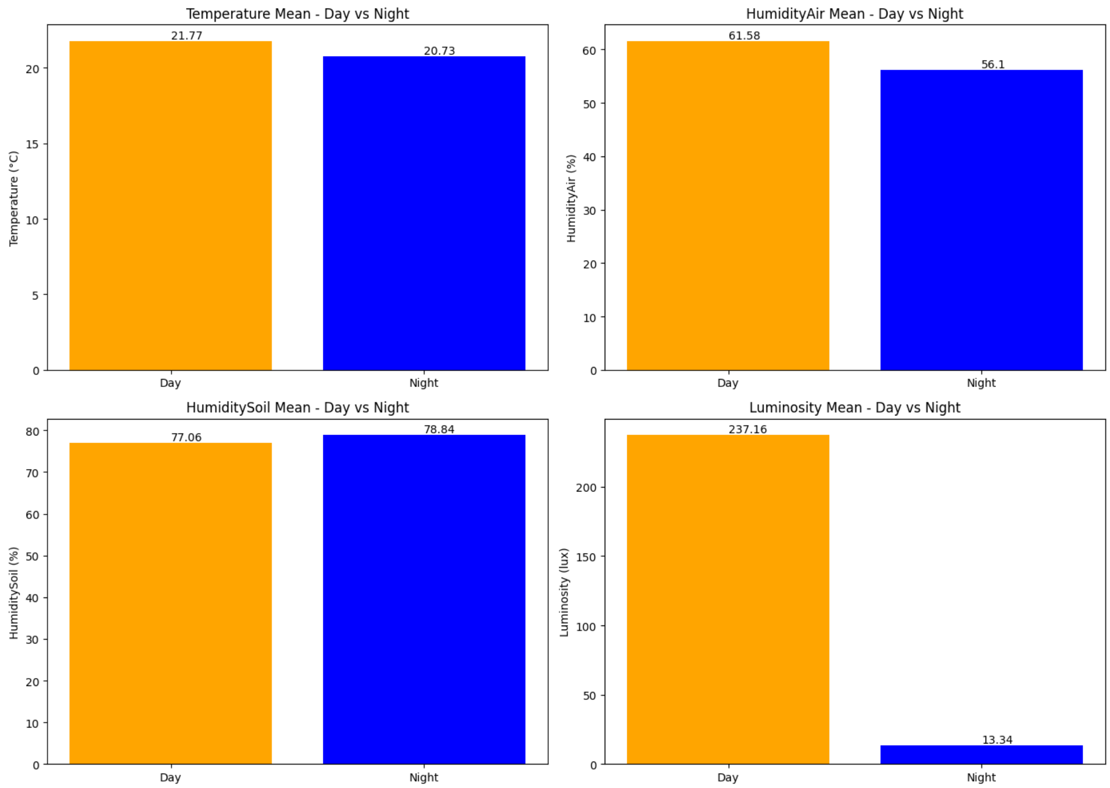


Figure 5.6: Greenhouse Environmental Variables - Cultivation Test (Day and Night Shifts).

The test results indicated an average humidity of 61.58% during the day and 56.10% at night.

As for soil humidity, an average humidity of around 75% was expected for both shifts, and the result obtained of 77.06% for the day and 78.84% for the night was very close to the expected, showing the efficiency of the system in irrigating the greenhouse internally.

The expected brightness between the day and night shifts would present a significant discrepancy, considering that during the day the lighting would be turned on at maximum and turned off at night. In Figure 5.6 this difference is clear, with an average illuminance of 237.16 lux during the day versus 13.32 lux at night.

Table 5.2 shows, in the second column, how long each actuator was on during the test days and, in the third column, the percentage in relation to the total test time that the actuator was on.

In relation to the LED strip, a value close to 50% of overall performance was expected, considering that it is on during the day and off during the night, each lasting 12 hours. An approximate value of 48% was obtained, showing that the LEDs were indeed activated properly.

According to the fan and heater times shown in the second column of Table 5.2, it is clear that the control system spent much more time trying to cool the interior than it did to heat it. This is due to the external temperature during the tests, which turned out to be higher than the defined setpoint values.

Actuator	Time Activate (h)	Total (%)
LED Strip	311.62	48.09
Ventilator	558.33	86.16
Water Pump	0.33	0.05
Heater	2.54	0.39
Atomizer	21.25	3.28

Table 5.2: Operation Time of the Actuators During the Cultivation Test.

In general, the evolution of all variables over the test days can be observed in Figure B.7.

Chapter 6

Conclusion and Future Work

This project addressed the development of an intelligent, automated, modular and indoor greenhouse for the production of microgreens aimed at domestic consumers. With the growing demand for healthier products, focused on nutrition and longevity, microgreens have gained prominence in the market. However, there is still no well-defined supply chain for these products. Microgreens are young plants known for their exceptional flavor and high nutritional content, making them ideal for restrictive diets. Therefore, a solution such as a greenhouse for the production of these microgreens proved necessary to help people interested in growing them, as these plants require specific care regarding temperature and humidity for healthy growth.

The system architecture is layered to separate the sensing and actuation part from the controller, network and greenhouse management system. This work focused mainly on the physical aspects, with hardware and software designed to control the greenhouse. All data was structured in a JSON file and published in specific topics using the MQTT protocol, widely used in IoT solutions. This approach ensures that the monitoring system is versatile and capable of receiving and sending information to the microcontroller via publications and subscriptions to a broker.

Regarding system control, the variables managed included temperature, air humidity, soil humidity, and illuminance. Actuators such as LED strips, a water pump, a set of fans, two heaters, and an atomizer were used. Each actuator directly impacts a specific variable; however, as discussed in Chapter 5, it was observed that heaters, e.g., influence not only the temperature but also the air humidity. Likewise, fans, by exchanging the greenhouse's internal air with external air, affect both the temperature and humidity of the environment.

Additionally, the flexible control system presented in Section 4.3.5, based on recipes for both day and night shifts, temporal checks for lighting control, hysteresis in air humidity and temperature control, and irrigation control timing, proved to be highly effective based on subsequent tests, as detailed in Chapter 5.

The tests carried out demonstrated the efficiency of the greenhouse in producing microgreens. In a period of 28 days, adding the germination time and the growth time, the system managed to grow young basil, although a little smaller than expected (about 2 cm instead of the expected 3 cm). The greenhouse prototype has limitations in temperature control, being able to increase the temperature by a maximum of approximately 10°C above the ambient temperature. On the other hand, refrigeration is limited by the temperature of the external air, relying only on a ventilation system and without a dedicated refrigeration system.

For future advancements, improvements may include installing additional sensors for more comprehensive coverage within the greenhouse, allowing for more complete data collection. Currently, only one sensor monitors each variable. Furthermore, optimizing lighting control with addressable LED strips could offer precise lighting control in different areas of the greenhouse. Exploring different wavelengths, such as ultraviolet light, can also be studied. Irrigation improvements may involve upgrading distribution systems for a more even distribution of water and properly managing pressure in irrigation pipes, addressing current challenges related to water pressure.

In conclusion, this system proved to be highly efficient in microgreen production tests and offers an intelligent solution to this problem. However, it is critical to address the identified limitations through future work for refinement and optimal performance.

Bibliography

[1] A. Ebert *et al.*, "Sprouts, microgreens, and edible flowers: the potential for high value specialty produce in Asia", *SEAVEG 2012: High Value Vegetables in Southeast Asia: Production, Supply and Demand*, pp. 216–227, 2013.

[2] M. Paraschivu, O. Cotuna, V. Sărățeanu, C. C. Durău, and R. A. Păunescu, "Microgreens-current status, global market trends and forward statements.", 2021.

[3] A. Bawa and K. Anilakumar, "Genetically modified foods: safety, risks and public concerns—a review", *Journal of food science and technology*, vol. 50, no. 6, pp. 1035–1046, 2013.

[4] M. C. Kyriacou, Y. Rouphael, F. Di Gioia, A. Kyratzis, F. Serio, M. Renna, S. De Pascale, and P. Santamaria, "Micro-scale vegetable production and the rise of microgreens", *Trends in food science & technology*, vol. 57, pp. 103–115, 2016.

[5] S. A. Mir, M. A. Shah, and M. M. Mir, "Microgreens: Production, shelf life, and bioactive components", *Critical reviews in food science and nutrition*, vol. 57, no. 12, pp. 2730–2736, 2017.

[6] M. T. El-Saadony, A. M. Saad, S. M. Soliman, H. M. Salem, A. I. Ahmed, M. Mahmood, A. M. El-Tahan, A. A. Ebrahim, T. A. Abd El-Mageed, S. H. Negm, *et al.*, "Plant growth-promoting microorganisms as biocontrol agents of plant diseases: Mechanisms, challenges and future perspectives", *Frontiers in plant science*, vol. 13, p. 923 880, 2022.

[7] G. Vox, M. Teitel, A. Pardossi, A. Minuto, F. Tinivella, E. Schettini, *et al.*, "Sustainable greenhouse systems", *Sustainable agriculture: technology, planning and management*, vol. 14, 2010.

[8] R. R Shamshiri, F. Kalantari, K. Ting, K. R. Thorp, I. A. Hameed, C. Weltzien, D. Ahmad, and Z. M. Shad, "Advances in greenhouse automation and controlled environment agriculture: A transition to plant factories and urban agriculture", 2018.

[9] R. Shamshiri and W. Wan Ishak, "A review of greenhouse climate control and automation systems in tropical regions.", 2013.

[10] D. Shirsath, P. Kamble, R. Mane, A. Kolap, and R. More, "IoT based smart greenhouse automation using arduino", *International Journal of Innovative Research in Computer Science & Technology*, vol. 5, no. 2, pp. 234–238, 2017.

[11] C. Gnauer, H. Pichler, C. Schmittner, M. Tauber, K. Christl, J. Knapitsch, and M. Parapatits, "A recommendation for suitable technologies for an indoor farming framework.", *Elektrotech. Informationstechnik*, vol. 137, no. 7, pp. 370–374, 2020.

[12] J. B. Baccay, C. P. Vicente, and M. T. Bravo, "IoT-based automated greenhouse with monitoring and control using MQTT protocol.", *Turkish Online Journal of Qualitative Inquiry*, vol. 12, no. 6, 2021.

[13] A. Badji, A. Benseddik, H. Bensaha, A. Boukhelifa, and I. Hasrane, "Design, technology, and management of greenhouse: A review", *Journal of Cleaner Production*, vol. 373, p. 133 753, 2022.

[14] A. Bhujel, J. K. Basak, F. Khan, E. Arulmozhi, M. Jaihuni, T. Sihalath, D. Lee, J. Park, and H. T. Kim, "Sensor systems for greenhouse microclimate monitoring and control: a review", *Journal of Biosystems Engineering*, vol. 45, pp. 341–361, 2020.

[15] S. Jin, S. Jingling, H. Qiuyan, W. Shengde, and Y. Yan, "A remote measurement and control system for greenhouse based on GSM-SMS", in *2007 8th international conference on electronic measurement and instruments*, IEEE, 2007, pp. 2–82.

[16] L. Li, K. Cheng, and J. Pan, "Design and application of intelligent control system for greenhouse environment", in *2017 7th international conference on power electronics systems and applications-smart mobility, power transfer & security (PESA)*, IEEE, 2017, pp. 1–5.

[17] B. Kang, D. Park, K. Cho, C. Shin, S. Cho, and J. Park, "A study on the greenhouse auto control system based on wireless sensor network", in *2008 International Conference on Security Technology*, IEEE, 2008, pp. 41–44.

[18] S. Alyousif, N. F. Zainuddin, and B. B. Hamzah, "Intelligent temperature control system at greenhouse", *International Journal of Applied Engineering Research*, vol. 12, no. 9, pp. 1811–1814, 2017.

[19] Y. Wu, L. Li, M. Li, M. Zhang, H. Sun, N. Sygrimis, and W. Lai, "Remote-control system for greenhouse based on open source hardware", *IFAC-PapersOnLine*, vol. 52, no. 30, pp. 178–183, 2019.

[20] W. Li, Q. Luo, Z. Li, and Y. Li, "The design and implementation of a low cost temperature control system for agriculture greenhouses", in *2009 International Conference on Energy and Environment Technology*, IEEE, vol. 1, 2009, pp. 399–401.

[21] S. Li, L. D. Xu, and S. Zhao, "The Internet of Things: a survey", *Information systems frontiers*, vol. 17, pp. 243–259, 2015.

[22] I. Yaqoob, E. Ahmed, I. A. T. Hashem, A. I. A. Ahmed, A. Gani, M. Imran, and M. Guizani, "Internet of Things architecture: Recent advances, taxonomy, requirements, and open challenges", *IEEE wireless communications*, vol. 24, no. 3, pp. 10–16, 2017.

[23] L. Zhang, I. K. Dabipi, and W. L. Brown Jr, "Internet of Things applications for agriculture", *Internet of things A to Z: technologies and applications*, pp. 507–528, 2018.

[24] X. Shi, X. An, Q. Zhao, H. Liu, L. Xia, X. Sun, and Y. Guo, "State-of-the-art Internet of Things in protected agriculture", *Sensors*, vol. 19, no. 8, p. 1833, 2019.

[25] M. B. Yassein, M. Q. Shatnawi, S. Aljwarneh, and R. Al-Hatmi, “Internet of Things: Survey and open issues of MQTT protocol”, in *2017 international conference on engineering & MIS (ICEMIS)*, Ieee, 2017, pp. 1–6.

[26] M. Glinz, “On non-functional requirements”, in *15th IEEE international requirements engineering conference (RE 2007)*, IEEE, 2007, pp. 21–26.

[27] D. Soni and A. Makwana, “A survey on MQTT: a protocol of Internet of Things (IoT)”, in *International conference on telecommunication, power analysis and computing techniques (ICTPACT-2017)*, vol. 20, 2017, pp. 173–177.

[28] Autodesk, *Tinkercad*, 2024. [Online]. Available: `https://www.tinkercad.com/`.

[29] E. Systems, *ESP32 Technical Reference Manual*, 2024. [Online]. Available: `https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf`.

[30] L. Aosong Electronics Co., *Dht22 digital-output relative humidity temperature sensor module*, 2007. [Online]. Available: `http://www.aosong.com/asp_showgoods.asp?id=114`.

[31] Keyestudio, *KS0510: Keyestudio TDS Sensor Module*, Accessed: 2024-06-20, s.d. [Online]. Available: `https://www.keyestudio.com/products/keyestudio-new-type-soil-humidity-sensor-module-for-arduino`.

[32] R. Semiconductor, *Bh1750fvi general description*, 2011. [Online]. Available: `https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf`.

[33] Keyestudio, *Gy-30 bh1750fvi digital light intensity sensor module*, 2011. [Online]. Available: `https://www.keyestudio.com/products/gy-30-bh1750-digital-light-intensity-sensor-module`.

[34] B. Matysiak and A. Kowalski, “White, blue and red LED lighting on growth, morphology and accumulation of flavonoid compounds in leafy greens”, *Zemdirbyste-Agriculture*, vol. 106, no. 3, pp. 281–286, 2019. DOI: `10.13080/z-a.2019.106.036`.

[Online]. Available: https://zemdirbyste-agriculture.test.deimantinis.lm.lt/wp-content/uploads/2019/08/106_3_str36.pdf.

[35] A. M. Omer, "Constructions, applications and the environment of greenhouses", *African Journal of Biotechnology*, vol. 8, no. 25, 2009.

[36] FIWARE, *Smart data models*, Accessed: 2024-06-18, 2024. [Online]. Available: https://www.fiware.org/smart-data-models/.

Appendix A

External Links

The codes developed in this work and other documents related to this work can be found at the following link: `https://github.com/lucaspinow/MasterIPB-Microgrens.git`.

Appendix B

Other Appendices

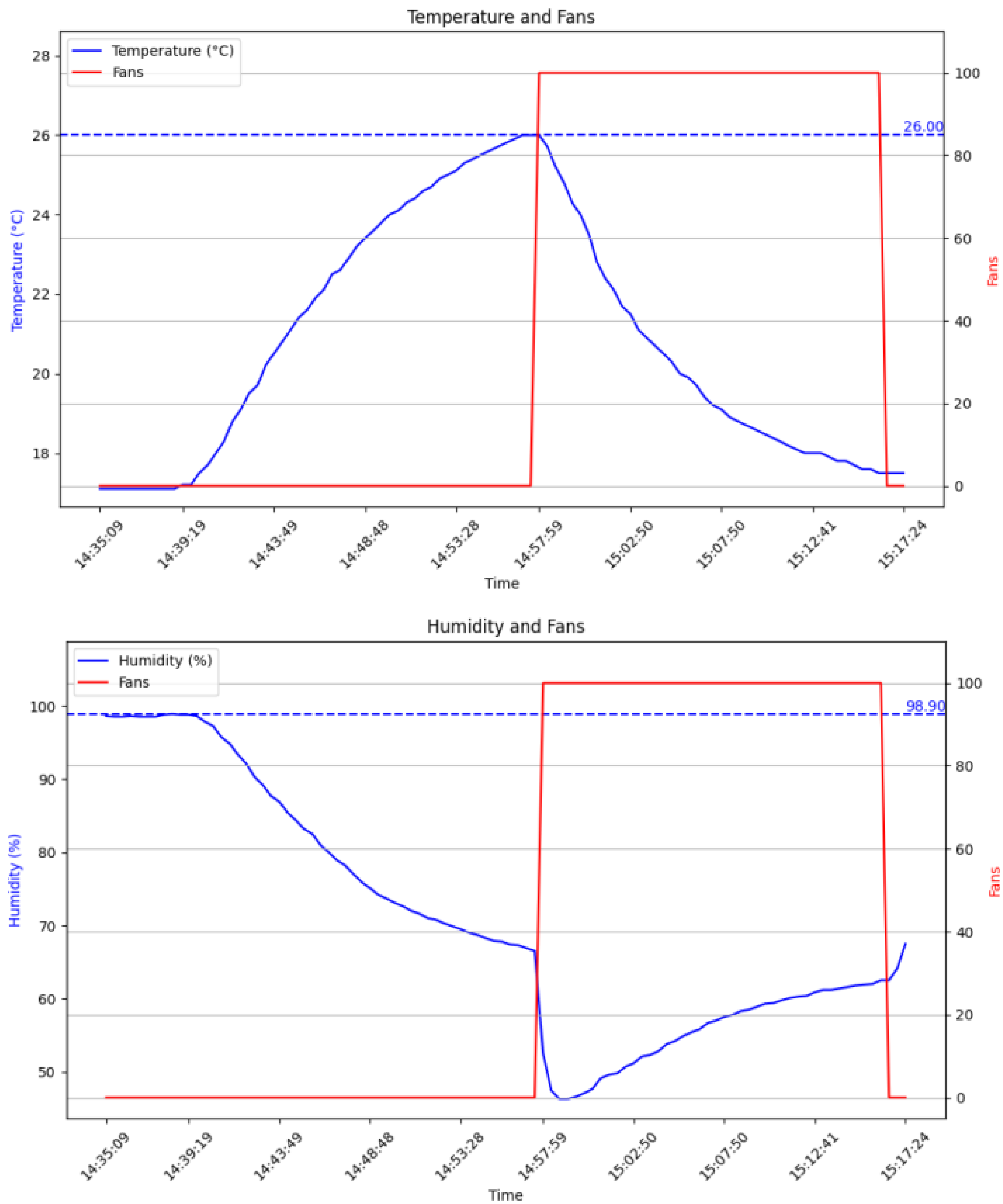


Figure B.1: Ventilator Influence - Temperature and Air Humidity.

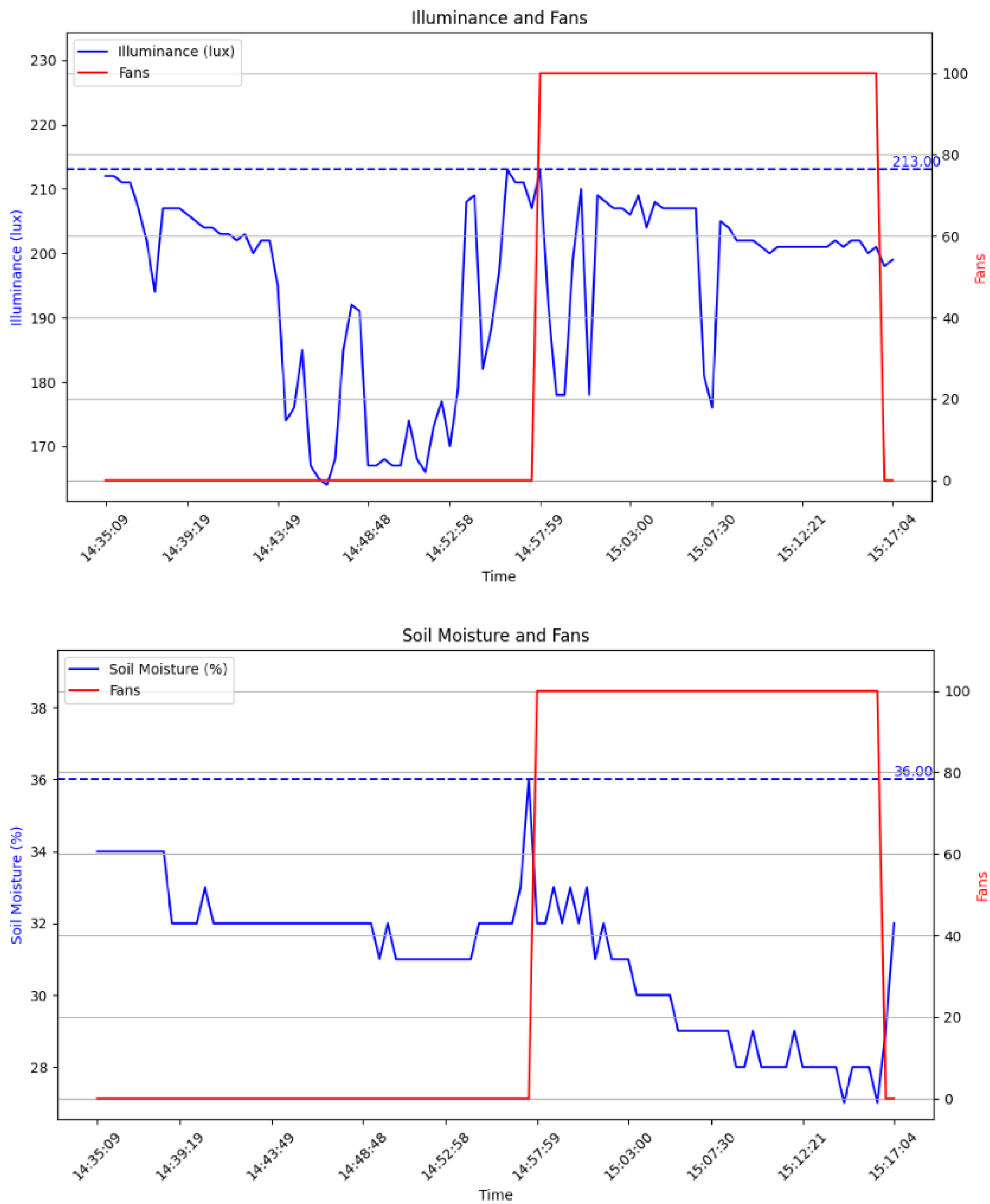


Figure B.2: Ventilator Influence - Illuminance and Soil Moisture.

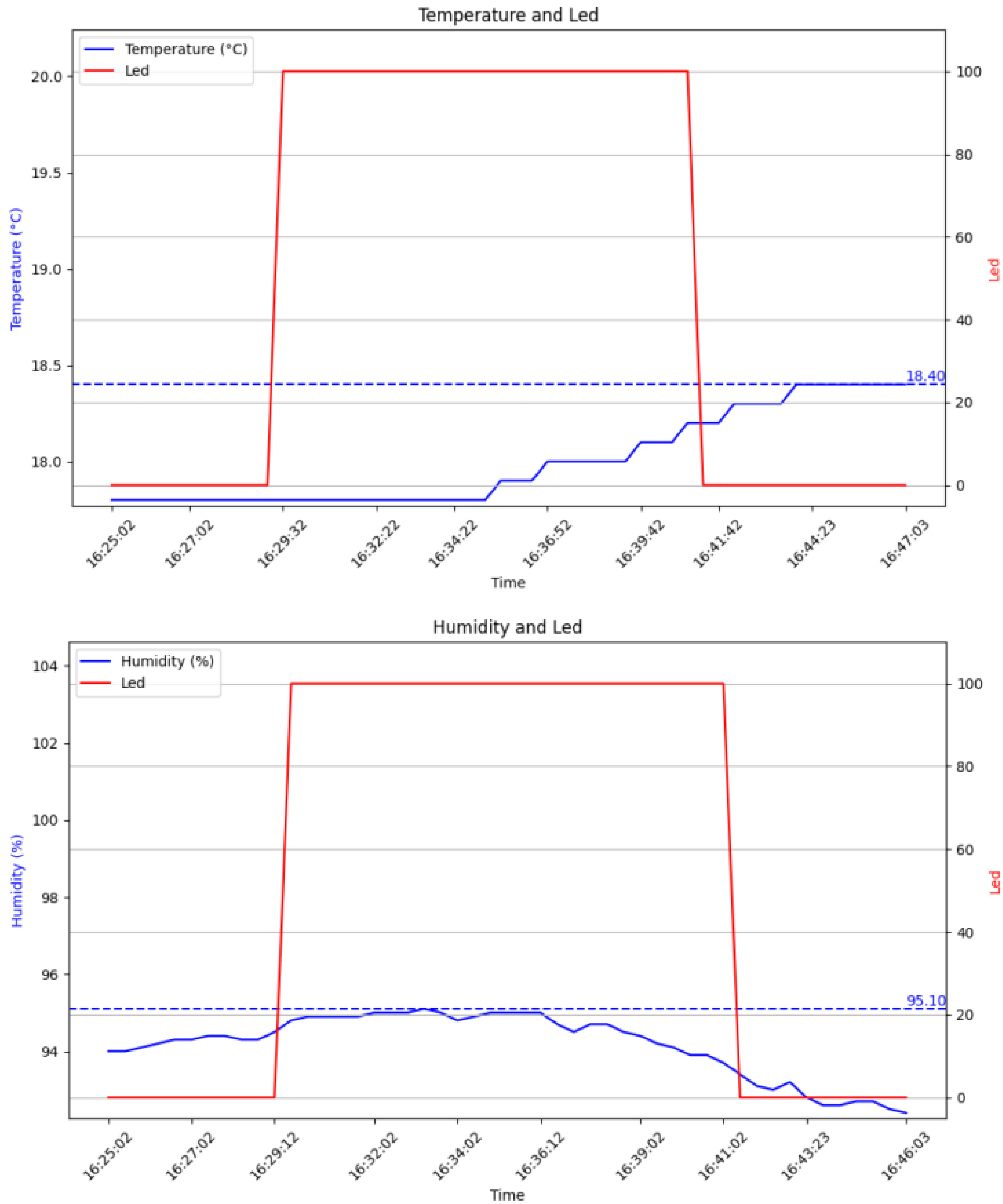


Figure B.3: LED Strip Influence - Temperature and Air Humidity.

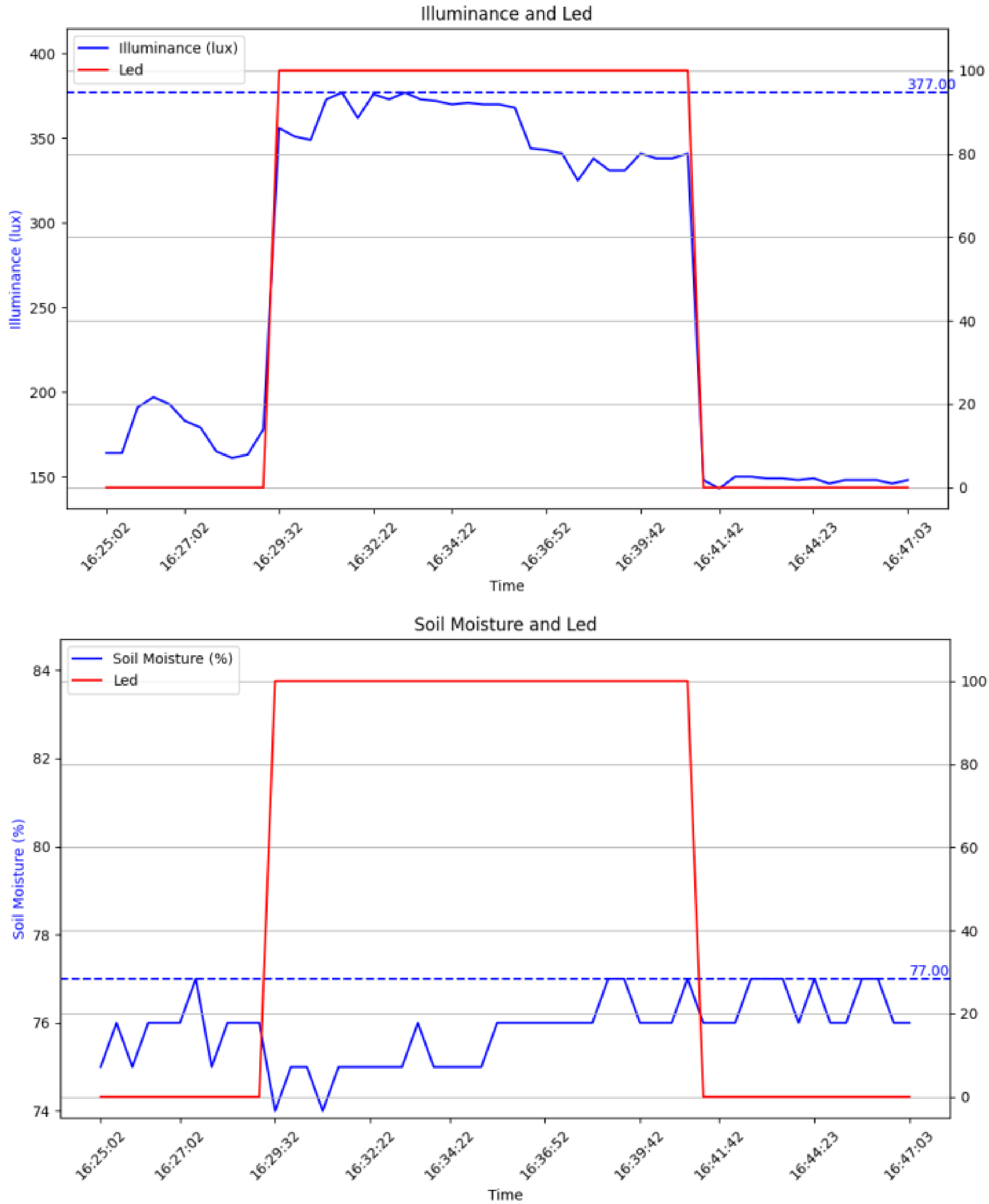


Figure B.4: LED Strip Influence - Illuminance and Soil Moisture.

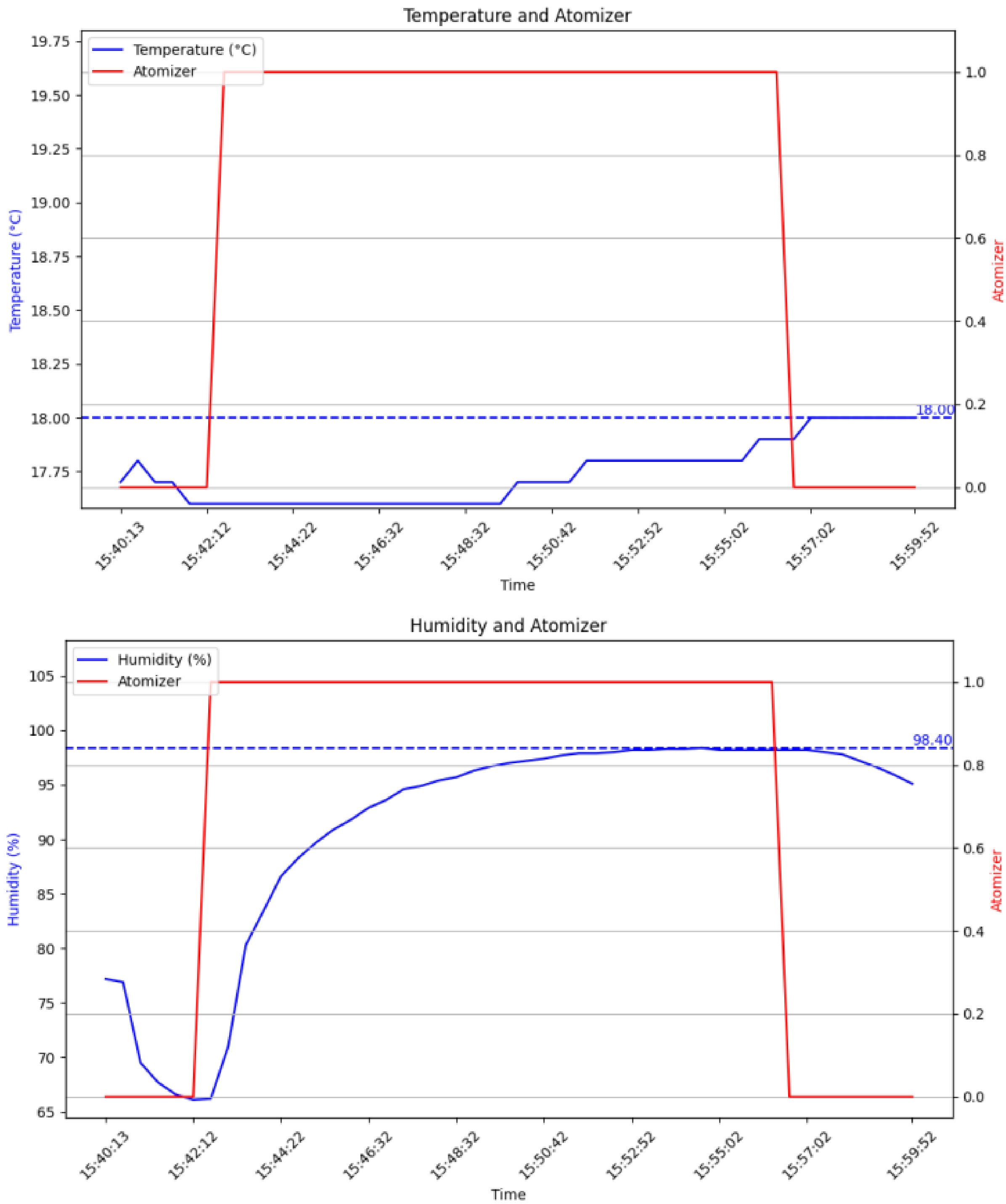
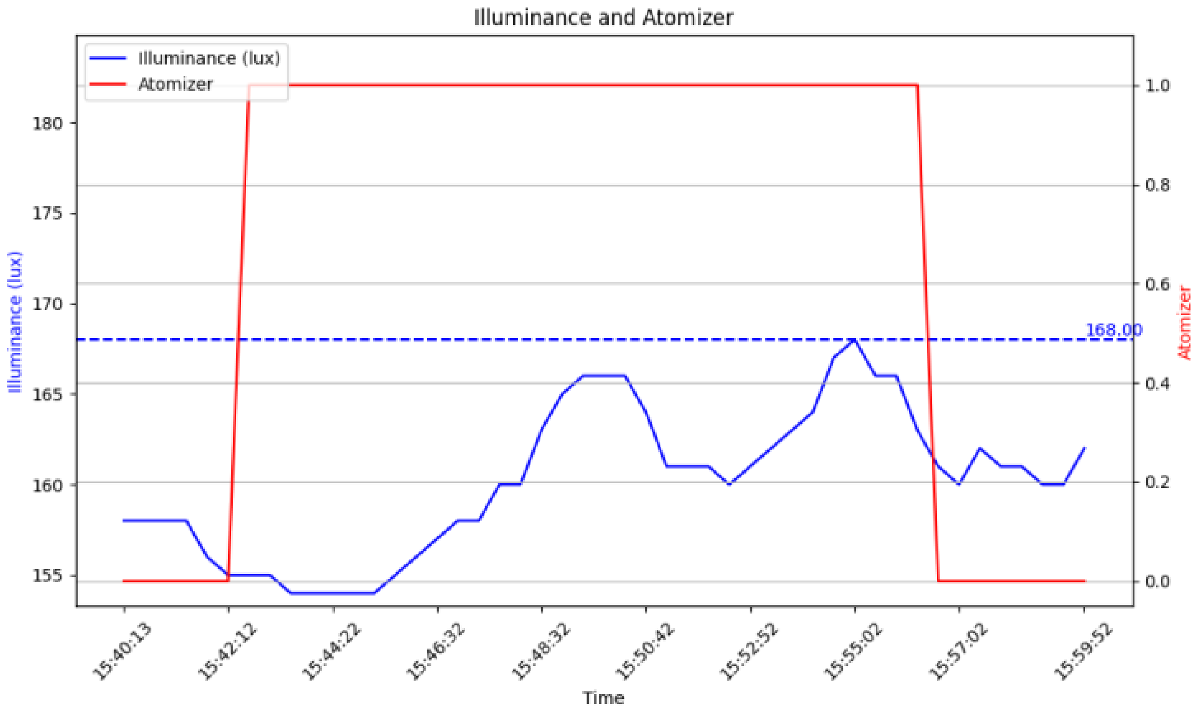


Figure B.5: Atomizer Influence - Temperature and Air Humidity.



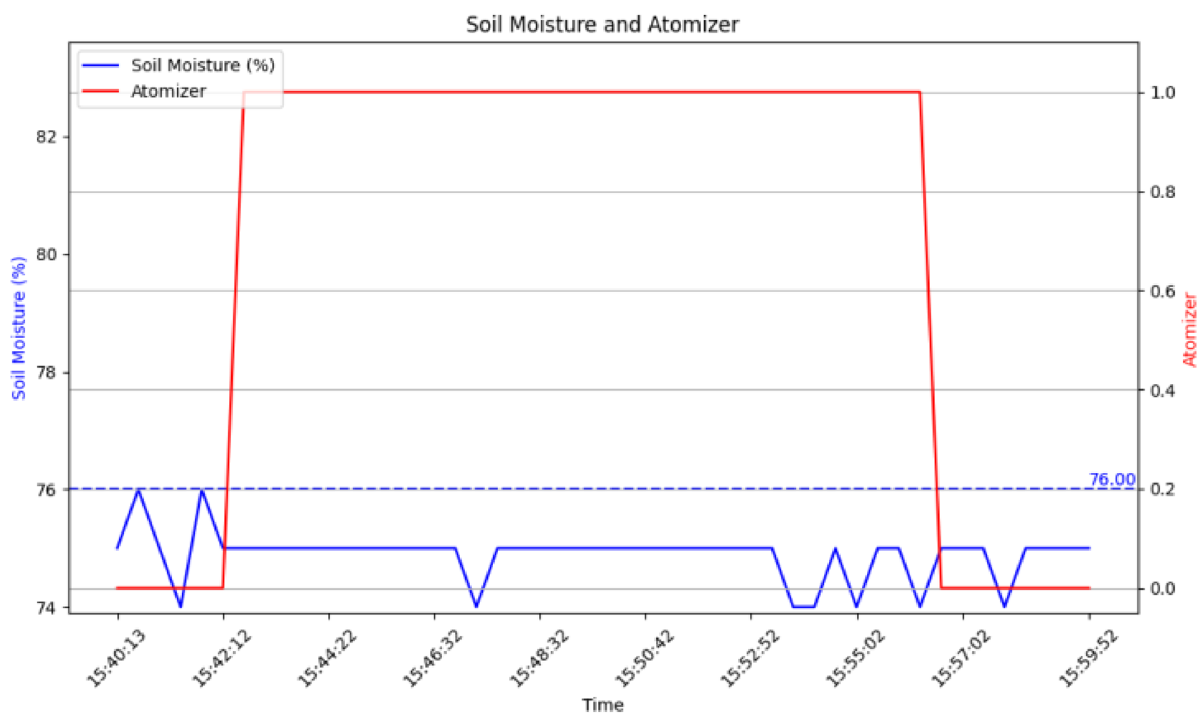


Figure B.6: Atomizer Influence - Illuminance and Soil Moisture.

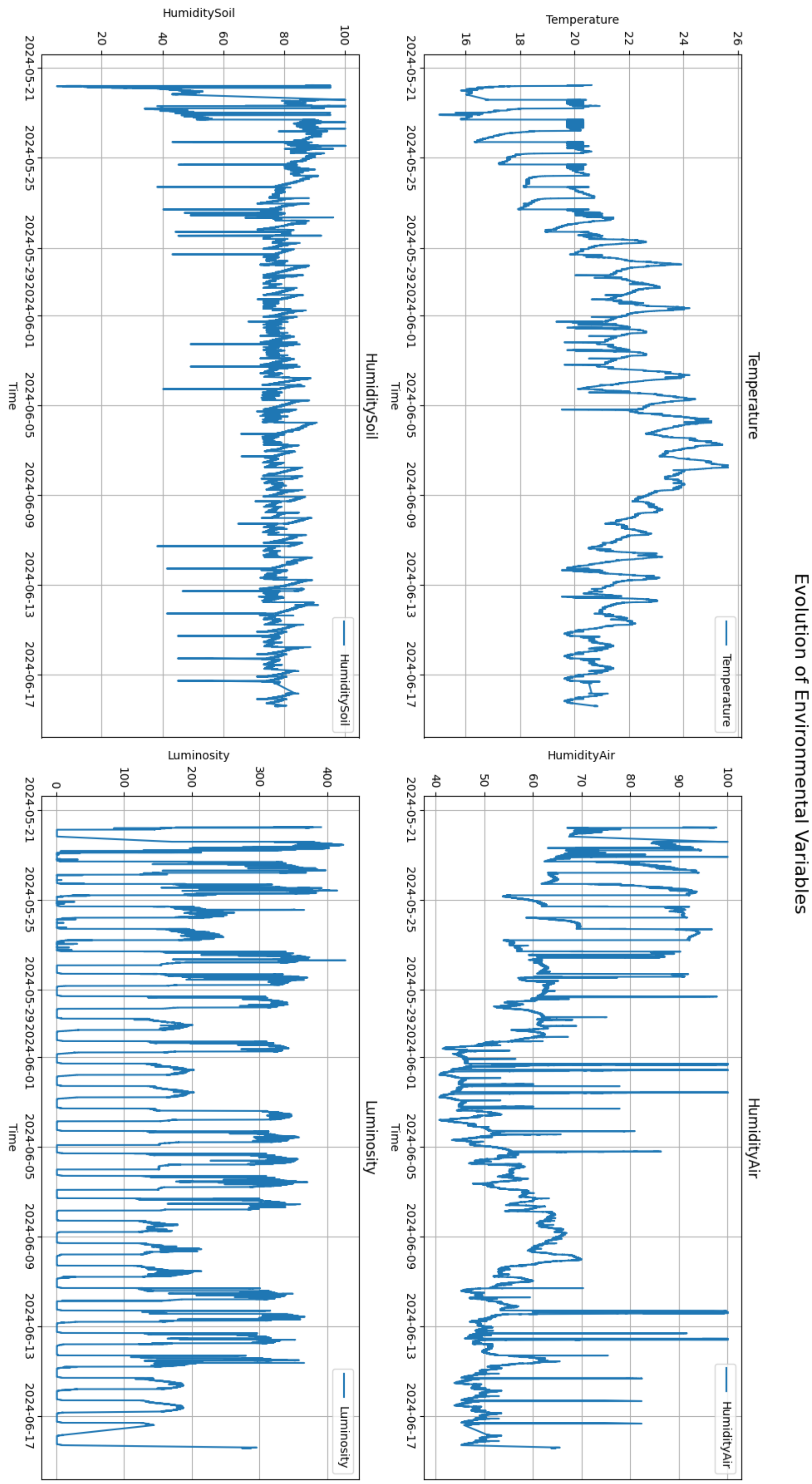


Figure B.7: Evaluation of Greenhouse Environmental Variables - Cultivation Test.