ipb
INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Tecnologia e Gestão

UTFPR
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Monitoring and Prediction System for Indoor Microgreens Growing Environments

Deborah Ishikawa da Silva

Work oriented by:
Prof. Paulo Jorge Leitão
Prof. Claudio Leones Bazzi
Prof. José Fernando Barbosa

Bragança
2023-2024

Monitoring and Prediction System for Indoor Microgreens Growing Environments

Deborah Ishikawa da Silva

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

I dedicate this work to everyone who supported me throughout its development, especially my mother Marlene and my supervisors Prof. Paulo Jorge Leitão, Prof. José Fernando Barbosa and Prof. Cláudio Leones Bazzi, who guided and encouraged me from the beginning.

Acknowledgement

Firstly, I would like to thank God who gave me the strength to complete this challenge. I would also like to thank my supervisors, Prof. Paulo Jorge Leitão, Prof. José Fernando Barbosa and Prof. Cláudio Leones Bazzi for the guidance and encouragement. To Lucas Santos Pinow who shared with me the challenges during the development of this work. To my family, especially my mother Marlene and my boyfriend Gabriel, who supported me on this journey. And to my friends who shared the moments with me during my double degree. You all helped me get here. Thank you all!

Abstract

Due to the increasing adoption of sustainable practices, urban agriculture has become more common. Microgreens, known for their rapid growth cycle, are particularly well-suited for development in these areas. However, traditional agriculture faces significant challenges throughout the cultivation process, such as instabilities, losses, and crop failures, primarily due to variations in environmental conditions that directly influence plant development. These difficulties are further accentuated in urban agriculture, as these areas typically lack adequate infrastructure and ideal climatic conditions. In this context, automated greenhouses emerge as a solution to stabilize cultivation by artificially replicating the necessary environmental conditions for plant growth, regardless of external conditions. This work explores using Internet of Things (IoT) technologies to monitor urban agriculture, enabling more predictable, adjustable, and optimized control of cultivation processes. In the analyzed scenario, integrating graphical interfaces for intuitive system monitoring allows for both local and remote tracking of the indoor environment, optimizing existing practices. Thus, continuous monitoring of parameters enables oversight, harvest forecasting, and decision-making, facilitating resource management during cultivation.

Keywords: Monitoring, automated greenhouses, crops prediction, technologies IoT.

Resumo

Devido à crescente inserção de práticas sustentáveis, a agricultura urbana tem se tornado cada vez mais frequente. Os microverdes, conhecidos pelo rápido ciclo de crescimento, são particularmente adequados para se desenvolverem nessas regiões. Entretanto, a agricultura tradicional enfrenta desafios significativos ao longo do processo de cultivo, como instabilidades, perdas e falhas de colheita, principalmente devido às variações nas condições ambientais que influenciam diretamente o desenvolvimento das plantas. Essas dificuldades são ainda mais acentuadas na agricultura urbana, pois essas áreas geralmente carecem de infraestrutura adequada e de condições climáticas ideais. Nesse contexto, as estufas automatizadas surgem como uma solução para estabilizar o cultivo, replicando artificialmente as condições ambientais necessárias para o crescimento das plantas, independentemente das condições externas. Este artigo explora o uso de tecnologias da Internet das Coisas (IoT) para monitorizar a agricultura urbana, possibilitando um controle mais previsível, ajustável e otimizado dos processos de cultivo. No cenário analisado, a integração de interfaces gráficas para a monitorização intuitiva dos sistemas permite o acompanhamento local e remoto do ambiente indoor, otimizando algumas práticas existentes. Desse modo, a monitorização contínua dos parâmetros possibilita a supervisão, previsão de colheitas e tomadas de decisões, facilitando a gestão de recursos durante o cultivo.

Palavras-chave: Monitorização, estufas automatizadas, previsões de colheita, tecnologias IoT.

Contents

1 Introduction **1**

1.1 Justification 2

1.2 Problem Statement 3

1.3 Objectives 4

1.4 Thesis Structure 5

2 State of Art **7**

2.1 Microgreens Crops 7

2.1.1 Environmental Conditions 8

2.1.2 Thermal Accumulation in Plants 10

2.2 Monitoring Automated Greenhouse 12

2.3 Internet of Things 14

2.3.1 IoT Architecture 15

2.3.2 IoT Applications 17

3 Methodology **19**

3.1 System Requirements 19

3.1.1 Functional Requirements 20

3.1.2 Non Functional Requirements 21

3.2 System Architecture 22

3.3 System Interface 25

4 Development **29**

4.1 Greenhouse Structure 29

4.2 Remote Interface Device 33

4.3 Cycle Change Prediction 41

5 Tests and Results **49**

5.1 Empty Tests 49

5.2 Plantation Results 56

6 Conclusion and Future Work **63**

List of Tables

4.1 Optimal conditions for microgreens [12][13]. 34
4.2 Daytime boundary conditions for microgreens [12][13]. 35
4.3 Night boundary conditions for microgreens [12][13]. 36
4.4 Microgreen harvest periods [12][13]. 44

List of Figures

2.1 Phenological scale of microgreens in relation to GDD (adapted from [16]). . 11
2.2 Traditional and cloud-based remote monitoring. 13
2.3 Comparation between the topologies of each protocol [22]. 16

3.1 Block diagram for preparing the physical architecture. 23
3.2 System architecture. 26

4.1 3D design of greenhouse architecture. 30
4.2 Base of the structure consisting of a tray and a perforated board, covered with an acrylic shield. 31
4.3 Front structure of the greenhouse. 32
4.4 Composition of actuators installed in the structure. 33
4.5 Atomization system installed. 33
4.6 Flow between nodes to establish revenues. 35
4.7 Flow between nodes for alerts depending on crop types and ideal conditions. 36
4.8 Control flow sent to MQTT. 37
4.9 Flow of sensor data sent by MQTT. 38
4.10 Flow of actuator states sent by MQTT. 38
4.11 Structuring some dashboards in the graphical interface. 39
4.12 Structuring the database for commands in a Fiware model. 40
4.13 Structuring the database for setpoints using a Fiware model. 40
4.14 Flow for reading data sets. 41
4.15 Structuring for forecast calculations and the possibility of changing the cycle. 42

4.16 Basil maturation conditions as a microgreen. 43
4.17 Calculation of GDH every hour. 44
4.18 Schematic flow for crop forecasts. 45
4.19 Different crop cycle options. 45
4.20 Schematic flow for combination of all conditions. 46
4.21 Crop forecast model located in the graphical interface. 46

5.1 First irrigation test. 50
5.2 Second irrigation test. 50
5.3 Testes de distribuição de água. 51
5.4 Resultados da distribuição de água. 51
5.5 Heating tests with thermal pads attached to heat sinks. 53
5.6 Air humidity tests using atomizers. 53
5.7 Air humidity tests using fans. 53
5.8 Warning when entering invalid values for cycle change. 55
5.9 User authentication system. 55
5.10 Cultivation of seeds in the substrate. 56
5.11 Environmental factors as a function of time during cultivation. 57
5.12 Monitoring of parameters in real time by Node-RED. 58
5.13 Fan performance over time during cultivation. 58
5.14 Heater performance over time during cultivation. 59
5.15 Performance of micro sprinklers over time during cultivation. 59
5.16 Atomizer performance over time during cultivation. 60
5.17 Remote monitoring by Raspberry Pi. 60
5.18 Local monitoring located on the module via OLED display. 60
5.19 First stages of basil growth. 61
5.20 Intermediate stages of basil. 61
5.21 Harvested basil microgreen. 61

Acronyms

CO_2 Carbon Dioxide. 64

3D Three-Dimensional. 29

API Application Programming Interface. 15

AWS Amazon Web Services. 18

CoAP Constrained Application Protocol. 17

CSV Comma-Separated Values. 39

FR Functional Requirements. 19

GDD Growing Degree Days. 10

GDH Growing Degree Hours. 43

HTML HyperText Markup Language. 39

HTTP HyperText Transfer Protocol. 16

IoT Internet of Things. 2

IP Internet Procol. 14

JSON JavaScript Object Notation. 39

L Liters. 30

LED Light Emitting Diode. 24

MQTT Message Queuing Telemetry Transport. 17

MySQL My Structured Query Languag. 40

NFR Non-Functional Requirements. 19

OLED Organic Light-Emitting Diode. 41

RFID Radio Frequency Identification. 15

V Volts. 30

WISP Wireless Identification Sensing Platform. 15

WSN Wireless Sensor Networks. 15

Chapter 1

Introduction

In recent years, urban agriculture has emerged as a response to the global challenges faced by conventional agriculture and sustainable practices. The rise of these farming practices within urban areas has been driven by the need to ensure practical access to fresh and nutritious food [1].

Conventional agriculture, on the other hand, is entirely dependent on environmental conditions for the ongoing development of crops. The impact of unpredictable conditions makes this situation critical, as it directly affects the quality of the produce. In contrast, urban agriculture provides a sustainable alternative, where growing conditions can be consistently controlled and stabilized [2].

In this context, the cultivation of microgreens stands out as an ideal practice due to their ability to thrive in small spaces and short growth cycles. These young plants, harvested shortly after germination, are known for their rapid growth. Moreover, microgreens offer continuous and flexible production that can be easily tailored to local needs [2].

The introduction of greenhouses in this setting provides an optimal environment for plant productivity and quality. Both traditional and automated greenhouses offer structures and systems conducive to successful cultivation. However, automated greenhouses enhance efficiency by optimizing the growing environment, resource use, and necessary handling, making them more effective compared to traditional greenhouses [2][3][4].

The integration of advanced technologies into cultivation systems has significantly contributed to managing all environmental parameters and ensuring stable production. The use of the Internet of Things (IoT) plays a crucial role in monitoring this environment by collecting real-time data from sensors and actuators. This data is then analyzed and adjusted according to the growing conditions, ensuring that microgreens thrive in an optimized and stable environment [2][3][4].

Automated greenhouses incorporate monitoring and control systems based on IoT, capable of creating specific microclimates that mimic the optimal conditions for microgreens' growth. These environments feature irrigation systems, controlled ventilation, and artificial lighting, enabling precise control over the factors that influence crop development [4][5].

Continuous monitoring of controlled environments through integrated IoT systems is configured to issue instant alerts regarding any deviations in environmental conditions, allowing for a swift response to potential issues. Additionally, remote monitoring enhances the predictability of crop development, providing insights and enabling proactive management [4][5].

Therefore, by adopting automated greenhouses and monitoring systems, growers can mitigate the risks associated with environmental variations and ensure an optimal environment for microgreens. This approach supports the development of crops in any setting and season, eliminating dependence on external environmental conditions [4].

1.1 Justification

The introduction of greenhouses has revolutionized agriculture by introducing controls over growing environments. These traditional structures help manually control internal conditions, providing a more stable environment protected from environmental impacts.

However, despite the promising use of greenhouses, they are still limited to human intervention to guarantee all control procedures. In this sense, the development of an automated environment makes it possible to emulate the environmental conditions required for each crop without dependence on these manual controls.

Integrating IoT architectures in greenhouses provides the interconnection of automated systems, enabling continuous monitoring of all parameters arranged in the cultivation system. This technology allows for a detailed, real-time view of the factors influencing the internal environment, optimizing previous mechanical supervision and control processes.

Continuous monitoring of these parameters ensures the detection of adverse conditions as well as the application of immediate adjustments when necessary. On the other hand, it is possible to configure recipes for each species of microgreen, guaranteeing personalized conditions during its development.

According to this information, farmers can measure harvest forecasts and optimize the growth cycle of microgreens according to their expectations. To this end, remote interfaces enable interaction with the user to obtain the desired information, integrate with predictive algorithms to check the possibilities of accelerating or delaying the end of cultivation, and, finally, apply readjustments to parameter controls.

In this way, all conditions can be manipulated according to the requirements defined by each crop and the producer's needs through remote interfaces. For this reason, controlled environments adjust environmental factors in the ideal proportions configured by the systems, changing the parameters to meet the conditions of the established cycles.

Crops grown in these conditions present more economical and sustainable practices without dependence on continuous human intervention, in addition to reducing waste in all processes. Cultivations from controlled environments result in high visual quality and more sophisticated products.

1.2 Problem Statement

Environmental conditions directly impact conventional agriculture and compromise the productive cycle during its vegetation phase. Continuous environmental instabilities hinder crop development, leading to changes in quality, harvest delays, water stress, and even yield losses.

Producers have introduced some alternatives to manage the imbalances in this scenario; however, they are generally limited to large-scale productions and are still insufficient to eliminate all negative effects. Concurrently, the demand for fresh food has increased with the rise in urban population, and the combination of these effects has intensified pressures on traditional agricultural systems.

Urban agriculture, in turn, has become increasingly common as an alternative to mitigate rural overload and ensure sustainable food production. Typically, crops are cultivated in small areas and environments with limited external conditions, requiring careful resource management to ensure sufficient crop supply.

However, regardless of the scenario, both conventional and urban agriculture face similar challenges related to environmental conditions. Despite being a sustainable solution, urban agriculture still relies on external conditions and must ensure measures that meet its fundamental needs.

Some conditions may be more favorable than others across different crops, over days, and in specific areas, and they can be unpredictable. Therefore, it is necessary to introduce technologies that enable the continuous performance of the agricultural system.

1.3 Objectives

This research's main objective is to develop a remote monitoring system for automated greenhouses, observing the behavior of crops located in indoor environments and analyzing possible changes to the cultivation cycle according to the producer's needs.

Monitoring systems should be able to switch between different microgreen crops, establish optimal conditions, estimate harvest forecasts, and switch between different metrics to maintain, delay, or advance cultivation. To achieve this, it is necessary to define some guidelines, such as:

- Develop an intuitive graphical interface for remote monitoring and implement a database to monitor historical data;

- Define optimal conditions for different microgreen crops and establish metrics that make it possible to change recipes depending on the crop to be produced;
- Establish limits in each condition to generate alerts when there are irregularities and analyze behavior according to these pre-defined conditions;
- Develop timers for the operating times of each actuator during cultivation;
- Establish live image transmission systems from inside the greenhouse to complement remote monitoring;
- Develop measurable crop forecasts and check possible changes to anticipate or extend the cycle;
- Allow interaction with the user to establish the cultivation cycle and the number of days they would like to change;
- Develop algorithms that adjust the parameters of previous conditions according to new user changes.

1.4 Thesis Structure

This document is structured into six chapters that contextualize the development of the work.

Chapter 1 comprises the introductory description of the research to understand its objectives and the reasons for its development.

Chapter 2 includes the state of the art composed of similar works developed in the same research area related to greenhouse development, crop handling, growth forecasting, system monitoring, IoT architecture integration, among others.

Chapter 3 presents the theoretical foundations of the research as well as the functional and non-functional requirements integrated into the architecture. This chapter covers the fundamental methodologies for implementing the system and integrating devices for connectivity, graphical visualization, and remote monitoring.

Chapter 4 covers all stages of architecture development, the system's graphical interface, and the calculations involving the thermal accumulation of each crop. These approaches were developed using IoT architectures and Node-RED. Additionally, this chapter introduces the security configuration implemented in the virtual environment where the system operates.

Chapter 5 presents the results obtained from tests conducted with the automated monitoring system. Empty tests and planting tests were conducted to analyze the system's behavior in both scenarios.

In Chapter 6, finally, conclusions are discussed along with the challenges encountered during the research development, as well as projections for future work.

Chapter 2

State of Art

This chapter provides an overview of technologies and research that have been developed with similar objectives to this work. It covers the solutions implemented, results obtained, and discussions on microgreen cultivation, automated greenhouse monitoring, and IoT applications.

2.1 Microgreens Crops

Microgreens represent a new category of crops originating from various plants such as vegetables, herbs, and wild plants. These crops have been deemed advantageous for many producers due to their increased nutritional value, despite requiring less maintenance and specific conditions compared to mature plants [6][7].

Additionally, the growth cycle of these microgreens allows for efficient, sustainable production within a short period of time. The harvesting period varies depending on the crop species, but generally, they are harvested around 7 to 21 days after germination [2][6][7].

They are considered young vegetables, consisting of a central stem and two pairs of immature leaves that are harvested before root development. When they reach an average height of 3 to 10 centimeters, they are harvested by individually trimming the shoots above the soil [6][7].

Despite these species existing for decades, the cultivation of microgreens has emerged with the development of controlled environments in urban agriculture. Crops grown in limited spaces within urban centers have arisen in response to demands for accessible, convenient, and nutrient-rich foods [6][8].

Microgreens demonstrate versatility in cultivation techniques, thriving in soilless substrates and standard soil-based methods. They adapt to various environments, enhancing economic factors in indoor agriculture. This production allows direct harvesting by consumers at any time, ensuring freshness for home-cooked meals [6].

In the agronomic scenario, three types of immature vegetables are consumed as functional foods. They are distinguished based on their size and growth stage: sprouts, microgreens, and baby greens. Sprouts are the youngest and smallest, consisting of seeds that have just germinated. Microgreens are slightly larger and intermediate in size. Baby greens have longer growth cycles and larger cultivation sizes [6][8].

All these crops function as functional foods composed of various nutrients. However, among the three microgreens, microgreens stand out for their ease and versatility of cultivation. They can be grown in limited spaces and have minimal environmental requirements, a rapid growth cycle, and continuous harvesting. They represent an intermediate stage between the other microgreens, making them more balanced in terms of nutrition and practicality [6][8].

2.1.1 Environmental Conditions

Microgreens exhibit flexibility in their cultivation systems, capable of being grown in organic and inorganic mediums as long as there is stable light, temperature, and humidity. The nutritional requirements for cultivating these crops are minimal compared to mature vegetables, making them ideal for urban agriculture [6].

Cultivation systems in controlled environments can be developed using various substrates, natural or artificial, and hydroponic systems. The substrate consists of a combination of solid particles, water, and air, which must be able to drain excess water to deeper layers [9][10].

Among cultivation systems, substrates in containers and blocks of porous materials stand out, composed of phenolic foams, rock wool, or coconut fibers. Both offer uniformity and ease of handling, making them widely used in controlled environments [9][10].

In general, regardless of the chosen substrate, it should include organic materials to facilitate aeration and supply oxygen, which are crucial for the nutrition of microgreens. For this reason, artificial substrates need to be supplemented with a layer of nutrient solution at the bottom of the trays [9][11].

Typically, the ideal scenario for cultivating these species involves temperatures ranging from 15 °C to 25 °C, with the optimal temperature being the average of both extremes. However, temperatures naturally fluctuate between day and night periods, which exhibit adaptive characteristics for plants.

During the day, temperatures fluctuate between maximums around 20 °C to 25 °C. Conversely, at night, temperatures are estimated to drop to around 10 °C to 15 °C on average [6][12][13].

This scenario accommodates the metabolic control of plants, favoring photosynthesis periods during the day and aerobic respiration during the night. Among other aspects, thermal factors directly influence all physiological activities during crop development [6][8].

Photoperiods are also fundamental factors for stimulating microgreens' growth, quality, and nutritional value. These crops are grown under exposure to light, whether natural or artificial, according to the length of the day corresponding to the seasons of the year. Typically, optimal light energy is provided for 12 hours within a 24-hour cycle [2][8].

As the substrate needs to remain moist, a minimum amount of water irrigation per day is necessary to avoid both excess and water scarcity throughout the cultivation period. Irrigation periods should ideally occur in the early part of the day to prevent the crop from remaining too wet overnight, as these conditions promote fungal growth [3].

Control systems vary based on soil moisture estimates, triggering irrigation when moisture levels drop below ideal and supplying just enough water to replenish moisture around the roots. Therefore, soil moisture can be kept consistent with daily irrigation [3].

Both air humidity and soil moisture can be detrimental in inadequate amounts during the microgreens production cycle. Therefore, regulating humidity is essential to minimize the likelihood of pest and parasite outbreaks. It is estimated that microgreens cultivation should maintain soil moisture levels around 70 % and air humidity between 70 % to 90 % [6][8][13].

There are various species that can be cultivated as microgreens, each with its own optimal environmental conditions and growth characteristics. Microgreens are commonly grown from vegetables and aromatic herbs belonging to the families Alliaceae, Apiaceae, Lamiaceae, Asteraceae, Chenopodiaceae, and Brassicaceae [12].

Generally, these species exhibit some morphological differences among themselves but require essentially the same optimal conditions to thrive. Allium vegetables belong to the Alliaceae family, Apiaceae includes carrots, parsley, and cilantro, Lamiaceae comprises mint and basil, Asteraceae consists of lettuces, Chenopodiaceae includes spinach and Swiss chard, and Brassicaceae comprises leafy greens like kale, watercress, and arugula. Thus, they encompass all the crops featured in this study [12].

2.1.2 Thermal Accumulation in Plants

In agronomic terms, the growth and development of plants are directly related to the accumulation of heat through the absorption of the average daily temperature. Thermal accumulation, defined as Growing Degree Days (GDD), represents a fundamental measure based on a base temperature that aids in modeling for forecasting and optimizing the crop cycle [14][15]. The Figure 2.1 represents the phenological scale of microgreens in relation to Degree-Days.

With technological advancements, the precision in calculating GDD has increased, benefiting monitoring in modern agriculture. Computational algorithms have enhanced the collection of real-time meteorological data, historical climate records, and specific information about each crop.

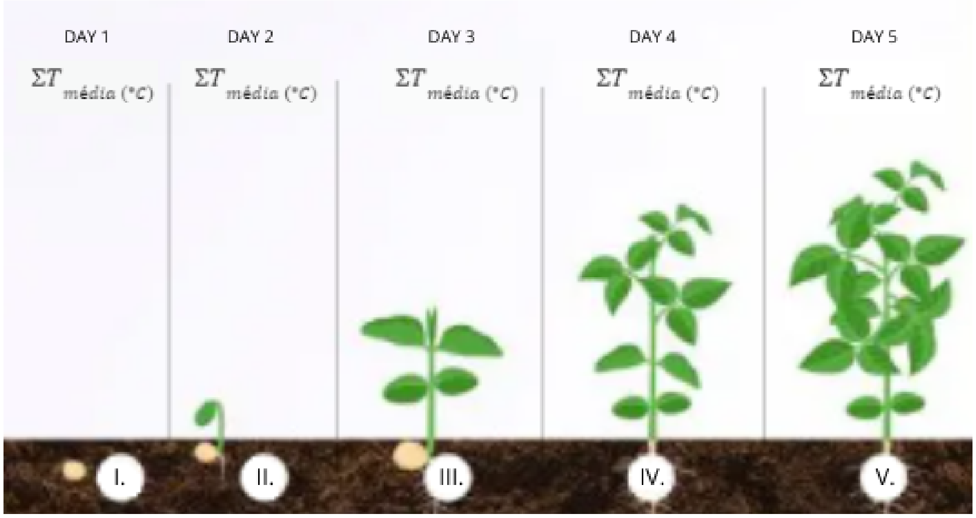


Figure 2.1: Phenological scale of microgreens in relation to GDD (adapted from [16]).

The GDD estimates not only enable quantitative forecasting of the production cycle but also have various other applications in agriculture. Based on these matrices, it is possible to plan planting and yield estimates aimed at maximizing crop performance during specific times of the year [14].

This parameter allows evaluating the impact of thermal conditions on plant growth through measurable indicators. Thermal accumulation throughout the crop development can be calculated by summing the following function [14].

$$GD = (\frac{T_{\text{max}} + T_{\text{min}}}{2}) - T_{\text{base}}$$

where T_{max} is the maximum daily temperature, T_{min} is the minimum daily temperature and T_{base} is the base temperature of a given crop. If the average between T_{max} and T_{min} is less than T_{base}, Degree-Days are zero [14][15].

Each crop also has a specific base temperature, which is the minimum temperature required for its development. Typically, the base temperature varies around 10 °C across different species [6][14].

While other environmental factors also influence the development of microgreens, temperature significantly favors crop growth, and GDD assists in applying simplified and measurable predictive models [6][14].

2.2 Monitoring Automated Greenhouse

Automated greenhouses include advanced control and monitoring technologies in order to minimize problems that may arise due to environmental imbalances. Continuous, real-time monitoring helps control ideal environmental conditions and optimize microgreen development [3].

Control and monitoring methods have been present in agriculture from the beginning to the present day, being revolutionized as technologies advance. Currently, there are already different methodologies implemented for monitoring automated greenhouses. However, the integration of multiple technologies for remote and real-time monitoring involves complementary systems for different functionalities [3][4].

Among the approaches to this technology, there are traditional and cloud-based remote monitoring. Both approaches enable remote monitoring of environmental conditions in greenhouses using sensors of critical parameters. However, traditional remote companies store their data on local servers, limiting system access. While cloud-based stores your data on external servers and can be accessed by any device connected to that server [4][17]. The figure 2.3 represents the comparative accesses between both approaches.

Both implementations have limitations that must be considered, such as traditional remote ones that have restricted scalability for the system expansion and dependence on local networks and data security systems. On the other hand, cloud-based monitoring solutions require constant internet connection, involve subscriptions for storage, and security systems are more sensitive by transmitting data to external servers [4][17].

Among them, the internal processes are basically the same based on information collected by the sensors. This information will be sent to a central system, which will be analyzed with the help of algorithms. These systems also automatically adjust environmental conditions to ensure they are always ideal for plant growth [4][17].

The main components of this monitoring include a) environmental and soil sensors,

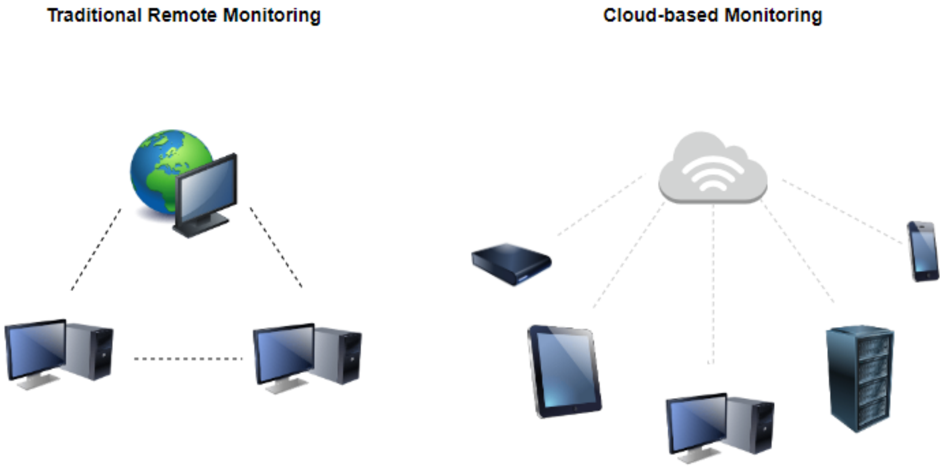


Figure 2.2: Traditional and cloud-based remote monitoring.

which constantly measure and report the internal state of the greenhouse; b) actuators, adjusting the environment according to the plants' needs according to the data obtained and; c) communication protocols, used to ensure that the exchange of information between sensors, central systems and actuators [4].

Environmental factors directly influence plant growth. Therefore, to monitor the optimal factors for each temperature, light, air humidity, and soil humidity condition, specific sensors are distributed in different positions in the greenhouse [4].

These sensors are connected to control systems that can activate the actuators responsible for making the necessary adjustments. The automation of these systems ensures that environmental conditions fall within ideal limits [4].

Among them, temperature instabilities can cause thermal stress, resulting in delayed growth, leaf burns, and reduced productivity. Regulations of air and soil humidity help prevent fungal diseases, water stress, or root rot. Furthermore, light also influences plant growth through processes such as photosynthesis and flowering [4][6].

Automated greenhouse systems also offer the advantage of remote access, allowing farmers to monitor and control their operations from anywhere via internet-connected devices. This is especially useful for managing multiple greenhouses or large-scale operations [4].

Therefore, the implementation of automated greenhouses results in higher quality production as well as more efficient use of resources, reducing waste and operational costs. With this, continuous monitoring provides an immediate response to inappropriate climate changes [4].

2.3 Internet of Things

In the past, each device operated in isolation and relied on human intervention to perform basic operations. The revolutionary emergence of digital interconnectedness in the modern world was driven by the convergence of various needs for automation and efficiency [18][19].

Due to technological advancements, IoT has experienced exponential growth, driven by a significant increase in electronic devices, the rise of Industry 4.0, and various communication technologies such as Bluetooth, Z-Wave, NFC, and Wi-Fi [18][19].

The concept of IoT refers to an environment in which almost all objects can be connected and communicate intelligently. In this context, sensors and actuators are integrated into physical objects and connect through networks, both wired and wireless, often using the Internet Procol (IP) [18][19].

The ability to transmit information over the internet is widely available in various devices today. These devices can perform intelligent reconfigurations, tracking, location services, real-time monitoring, and process control. In recent years, there has been a significant increase in the number of devices capable of connecting to the internet [18].

There are three essential components in an IoT architecture: a) hardware, consisting of sensors, actuators, and other embedded communication devices, b) middleware, which includes storage solutions and computing tools, and c) presentation, involving data visualization and interpretation tools [18].

Generally, this concept is described as a flexible and adaptive global network infrastructure with self-configuring capabilities, using communication standards and protocols. In this context, IoT constitutes a network of physical and virtual objects with their own identities that interact through intelligent interfaces without human intervention [19].

For this reason, it is necessary to develop the integration of multiple technologies that enable the identification and communication of objects. Thus, there are systems such as Radio Frequency Identification (RFID), Wireless Sensor Networks (WSN), Wireless Identification Sensing Platform (WISP), among others, used for individual object identification [20].

2.3.1 IoT Architecture

The architecture of an IoT system must ensure complete integration between the physical and virtual worlds. Considering that devices can move and require real-time interactions, the IoT architecture must be flexible and adaptable, allowing devices to interact dynamically [19].

This technology consists of sensing, networking, and application layers of technology. At the detection layer, systems are able to identify and collect data from connected devices. As each device has a unique digital identity, it is possible to monitor and track these devices in the digital environment [18][19].

Next, the network layer connects all devices, enabling them to recognize and interact with the environment around them. Devices exchange information with each other for efficient management and processing in the IoT environment. At this layer, information confidentiality and human privacy are critical as IoT connects a lot of private information [18][19].

Finally, the top of this structure contains the services layer, where the processed data is used to run various applications. This layer is made up of a set of Application Programming Interface (API) and protocols that support the services necessary for the system to function [18][19].

In IoT, compatibility between devices covers information exchange, communication and event processing. Therefore, it is necessary to compose an interface mechanism that facilitates the management and integration of connected devices [19].

Data acquisition in IoT comprises traditional data acquisition and IoT-specific data acquisition. Traditional data acquisition uses communication protocols such as SIGFOX, Wi-Fi and ZigBee, which are used for remote and short-range communication [21].

IoT-based data acquisition mainly uses two protocols, namely Client-Server and Publish-Subscribe. The Client-Server model is a classic network communication pattern, where one device acts as a client and another as a server. In this model, the architecture is centralized and supports direct communication between client and server, whose scalability is limited. On the other hand, the Publish-Subscribe model involves indirect communication, in which publishers send messages to subscribers mediated by a broker, allowing greater flexibility and scalability [4][22].

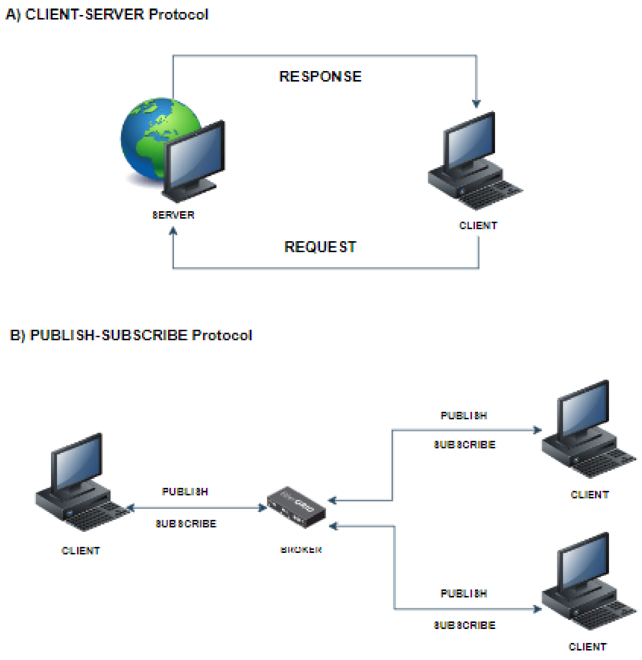


Figure 2.3: Comparation between the topologies of each protocol [22].

Figure 2.3 illustrates the differences between the topologies of the protocols presented. Among them, IoT systems define protocols according to the needs of each application. In this context, the present work depends on greater flexibility in its systems, being more compatible with the use of the Publish-Subscribe model.

To implement these protocols, it is necessary to use HyperText Transfer Protocol

(HTTP), Constrained Application Protocol (CoAP) and Message Queuing Telemetry Transport (MQTT) technologies, designed to facilitate communication in IoT systems [21][22].

HTTP and CoAP are compatible technologies for implementing the Client-Server protocol, mainly used for transferring data on the web. HTTP supports text-based application communication, while CoAP uses compact binary messages [21][22].

MQTT, in turn, is a technology compatible with the implementation of the Publish-Subscribe protocol, designed for real-time monitoring and remote control of devices. This technology supports lightweight communications in remote locations [21][22].

IoT devices communicate using specific protocols. Thus, the IP defines a set of rules that regulate data transmission over the Internet. These protocols ensure that information from a device can be understood by other devices, gateways or services [21].

Finally, the application layer acts as the point of interaction between the user and the device in a specific IoT protocol for these communications, such as MQTT [21].

2.3.2 IoT Applications

IoT applications are dynamic in a variety of sectors, such as a) smart homes, remotely controlling security, lighting, temperature and entertainment systems; b) in health, enabling real-time monitoring of patients and medication management; c) in industry, improving manufacturing operations, managing inventories and optimizing maintenance; d) in agriculture, continuously monitoring environmental conditions during crops; e) among others [20].

Among them, this technology is revolutionizing the agricultural sectors with the monitoring of environmental conditions. All this information facilitates decision-making, resource management, and environmental preservation. To optimize crop conditions, it is essential to collect data and methods that allow predicting future climate conditions based on historical data [5][20].

In this context, the WSN are strategically distributed within the greenhouse to facilitate monitoring and control of devices. These networks capture data using specific communication protocols, such as Publish-Subscribe via MQTT technologies [4].

The IoT platform provides an automated greenhouse environment capable of collecting, processing and interpreting data in real time. The system designed to automate both monitoring and control, sends captured information to the IoT Cloud, allowing easy and remote access at any time [4].

There are several platforms that facilitate this integration, offering everything from programming tools to environments for user interfaces, such as Amazon Web Services (AWS), IBM Watson, and Node-RED. Among them, Node-RED stands out for being a flexible platform for open-source development [4][23].

In this scenario, the Node-RED platform facilitates the integration of devices and services through visual programming, enabling remote monitoring of crops in indoor environments. IoT applications enable mechanisms that optimize system operations and increase crop quality [4][23].

Chapter 3

Methodology

This chapter addresses the methodology employed to achieve the proposed objectives, specifying both functional and non-functional requirements, the operational context of the system, technologies and architectures, as well as the ideal elements for the development of cultivation.

3.1 System Requirements

According to Sommerville (2011), system requirements are established to achieve specific project objectives. These requirements are divided into two main categories: Functional Requirements (FR) and Non-Functional Requirements (NFR). [24]

Functional requirements describe what the system should accomplish. They vary depending on the type of system being developed, the requirements, and the adopted development methodology. Thus, functional requirements define the functionalities and structure of the system.

On the other hand, non-functional requirements do not directly refer to the specific services that the system provides. They establish the context and constraints under which the system must operate, including aspects such as performance, security, and usability.

3.1.1 Functional Requirements

Functional requirements specify the actions and architecture of the system, divided into architectural functional requirements and interface functional requirements.

The functional requirements of the system architecture encompass parameters related to the efficient operation of the system through device networks and scalable structuring, including:

- FR01: The structure must be sized to ensure sufficient space for growth and adequate air circulation;
- FR02: The system architecture should accommodate sensors and actuators to monitor and control the necessary environmental conditions;
- FR03: The system must include a water reservoir of adequate size for the cultivation requirements;
- FR04: The greenhouse cover should be movable to facilitate maintenance and handling of the modules;
- FR05: The support structure should consist of perforated material capable of draining excess water to deeper layers;

While the functional requirements of the graphical interface encompass the data to be collected, the system's operational method, the visual interface, and monitoring, including:

- FR06: The interface should display real-time information about the crop status, such as temperature, air humidity, soil moisture, and luminosity;
- FR07: The interface should include the operating times and the status of each actuator located in the greenhouse;
- FR08: The interface should enable remote adjustment of operational parameters of the system, such as irrigation, diffusion, lighting, heating, and ventilation;

- FR09:The system should include automatic and manual operation modes for controlling devices;
- FR10: If the automatic operation mode is enabled, the control panels and setpoints must be disabled;
- FR11: The system must provide a list of options for various crops to define specific growing conditions;
- FR12: The interface should issue alerts if there are critical conditions according to the established operating limits;
- FR13: The interface should include a button to initiate cultivation, thereby triggering all subsequent operations of the system;
- FR14: The system should present growth forecasts through charts and indicators;
- FR15: The system should provide different predefined operation modes to maintain, delay, or accelerate the crop cycle.

3.1.2 Non Functional Requirements

Non-functional requirements define the criteria that determine the system's operation, including performance, security, and usability, divided into non-functional requirements for architecture and non-functional requirements for the interface.

The functional requirements of the system architecture encompass parameters related to the system's operation considering security, efficiency, and reliability, including:

- NFR01: The placement of sensors and actuators within the greenhouse must be installed in strategic positions to ensure operational efficiency;
- NFR02: The system must support the expansion of the infrastructure without significant changes to the existing system;

- NFR03: The system must be designed to minimize the energy consumption of operational devices such as fans, heaters, lighting systems, and irrigation;
- NFR04: The architecture must be dimensioned to support the structural load, equipment load, wind load, and the weight of the crop itself;

While the non-functional requirements of the graphical interface encompass efficiency, security, intuitive modes of system operation, visual interface, and monitoring, including:

- NFR05: The system must store historical data in a database for analysis throughout the crop development;
- NFR06: The interface should include intuitive charts and control panels for user interaction;
- NFR07: All elements of the interface must be clearly identified and understandable;
- NFR08: The system must be highly reliable, ensuring continuous and stable operations;
- NFR09: The generated alerts must be displayed on the visualization platform;
- NFR10: The graphical interface should be tailored for IoT applications and be easily reconfigurable;
- NFR11: The system must be capable of logging the date and time throughout the data acquisition process;
- NFR12: The system must support communication protocols, such as MQTT, to facilitate integration with IoT devices and the graphical interface.

3.2 System Architecture

Due to the challenges faced in conventional agriculture, the project aims to monitor a controlled environment for the cultivation of microgreens that will be developed to mitigate the effects of climate instability. Therefore, this topic involves the methodology for constructing a controlled greenhouse.

According to Figure 3.1, this work implements features in the cloud based on data processed at the edge, aiming to optimize the resources used. Controlled environments must support all operational systems to perform the necessary conditions for each crop and make harvest forecasts.

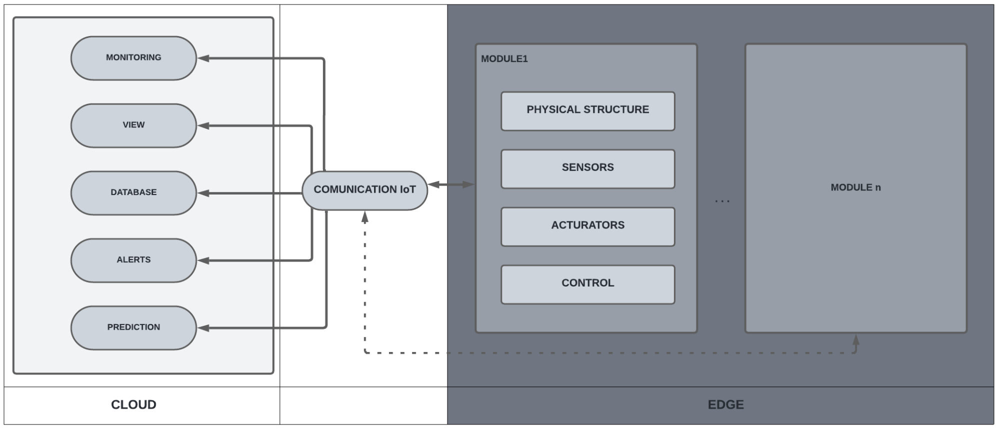


Figure 3.1: Block diagram for preparing the physical architecture.

Greenhouses are structures designed to manipulate environmental factors, providing favorable conditions for the productive development of plants. The system architecture basically comprises a support structure and a transparent cover sized to support the structural, equipment, and cultivation load. [25].

Dark coverings not only block the external visibility of the cultivation but also act as heat accumulators, leading to undesirable effects due to increased internal temperatures. Therefore, the covering should be transparent or light-colored and made from materials such as plastic, glass, or acrylic. Meanwhile, the support structure can be constructed based on cost-effectiveness from materials such as metal, wood, or plastic [25].

Therefore, the proposed system must include a modular structure and a transparent mobile cover consisting of ventilation, heating, irrigation, atomization, lighting, and sensors for monitoring systems [25]. The architecture of the proposed system presents the block diagram structured by components and functionalities, as shown in Figure 3.1.

The irrigation method must be established according to available water resources and production species. In this context, the conduits will be supplied from a water reservoir.

As compared to the external environment, air circulation in the indoor environment presents thermal inversion phenomena inside due to evapotranspiration, it is necessary to maintain constant air exchange. The ventilation modules help exchange air between the internal and external environment in order to balance the perspiration and absorption of microgreens. For this reason, the proposed ventilation systems must be composed of pairs of exhaust and circulation fans [6][25].

Air temperature and humidity controls are essential to minimize physiological disorders in crops. To do this, a sensor must be installed that continuously monitors air humidity conditions as well as temperature to be monitored in the Cloud [26][27].

Light Emitting Diode (LED) devices locate artificial lighting in the modular structures of this project. These devices, in addition to presenting a controlled light spectrum throughout the greenhouse, place wavelengths favorable to stimulating growth and nutrient accumulation in microgreens [6][27].

Generally, the ratio of blue to red light consists of 20 % to 80 % in which blue light benefits the early stages during germination and red light, in turn, stimulates the formation and absorption of photosensitive pigments, responsible for plant growth [6][8][27]. For this reason, the proposed system must include LED strips with a combination of red and blue lights distributed across the top of the greenhouse. Furthermore, the architecture must include a light-intensity sensor to aid monitoring.

All components must be integrated from a central circuit to control the ideal conditions during the cultivation of microgreens, being monitored, predicted and visualized in the Cloud. Finally, the sensors must be distributed at strategic points for efficient acquisition of the parameters monitored in the Cloud, in addition to comprising a compact camera to capture images in real-time.

3.3 System Interface

The proposed system must ensure the integrity of controlled environments through a system consisting of a graphical monitoring and control interface. The monitoring modes should oversee all parameters within the greenhouse to track the cultivation's performance [4]. To achieve this, specific programs will be developed on a central server using Node-RED, which will manage data flows and visual interactions.

The graphical interface should be accessible remotely via IoT applications, displaying real-time data and allowing control adjustments in a visually intuitive manner [4][19]. The parameters that will be visualized include the temperature, light intensity, air humidity, and soil moisture, as well as the statuses of the actuators within the greenhouse, as illustrated in Figure 3.2.

According to Figure 3.2, it is possible to observe that this work performs functionalities in the management and user interface layers. The management layer includes monitoring, recipe definition, visualization, and prediction of parameters, while the user interface layer displays all information in real-time via local or remote interfaces.

The operating system will be organized into specific niches, displaying all parameters through historical and real-time graphs. Moreover, the system's visual architecture will include video integration to complement real-time monitoring during the cultivation process. This setup will provide a comprehensive overview of the system and enhance the understanding of the current state of the microgreens.

The monitoring system must allow for the selection of different crops, adjusting the parameters to meet the specific needs of each one. This functionality supports different recipes for each crop, configured in the management layer. Additionally, the interface should enable the monitoring and control of all sensors and actuators within the greenhouse.

The operation modes will alternate between automatic and manual, configured remotely by the user through the graphical interface. When the system is in automatic mode, it will be set to the optimal conditions for the selected crop without allowing

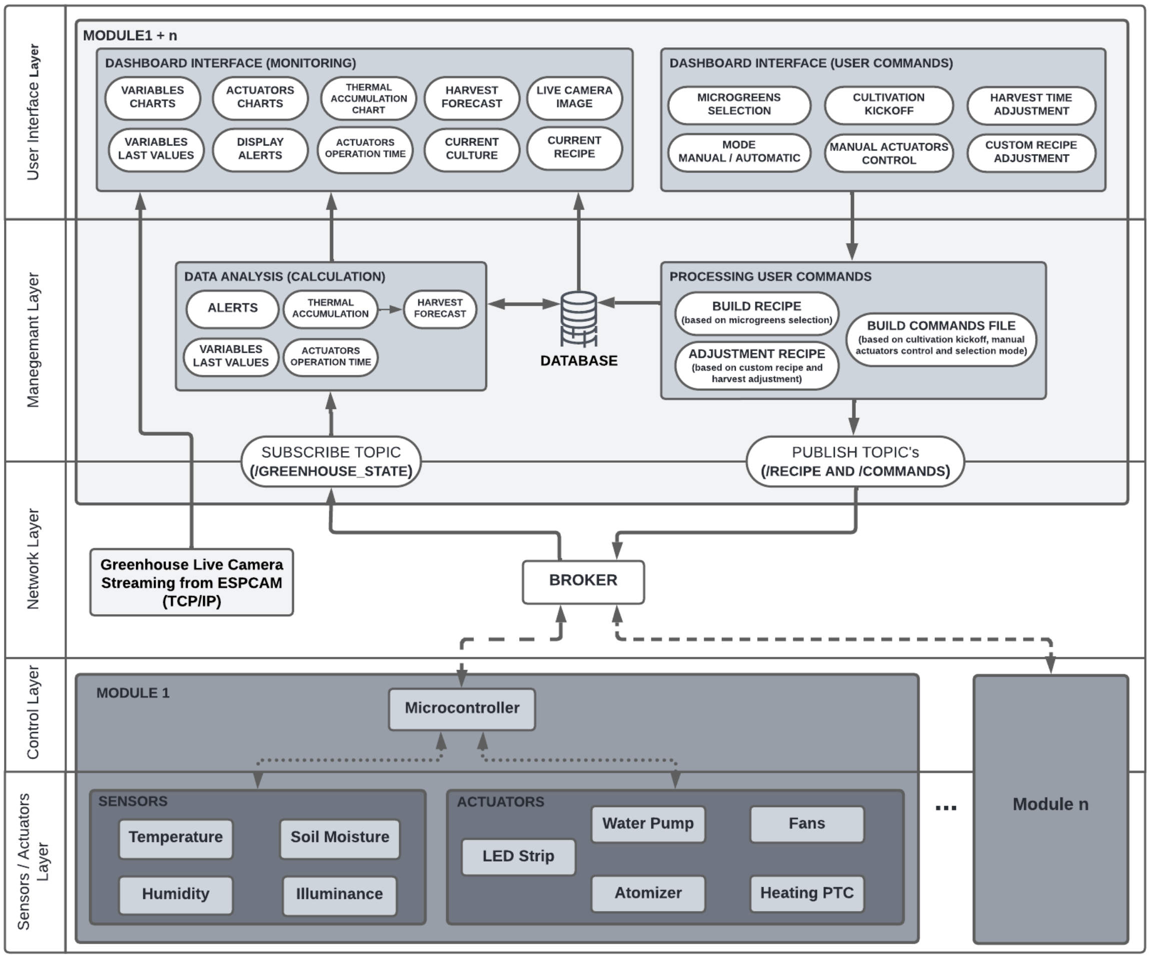


Figure 3.2: System architecture.

manual adjustments.

On the other hand, all parameters must remain within their optimal limits, and the system should always issue alerts when there are significant deviations from these established parameters. All these characteristics and functionalities present in the management architecture and user interface are located in Figure 3.2.

The system should also monitor and record the operating time of all active devices, such as irrigation valves, atomizers, fans, heaters, and lighting systems, from the start of cultivation until harvest.

Monitoring factors can estimate the time and growth progress of microgreens based on collected data. The system should be capable of predicting the end of cultivation based on the thermal accumulation of microgreens during growth. Subsequently, the system can assess options to advance or delay cultivation according to the producer's expectations.

Finally, monitoring must include a storage database of historical data and system configurations. Databases will be managed according to the data processed in the management layer and read when necessary to resume rebooted systems. Access to the graphical interface must be restricted to authorized users and must support local and remote access on any device, using only the IP address.

Chapter 4

Development

This chapter discusses the key components and functionalities for structuring the greenhouse, remote interface devices, and crop cycle forecasting. These systems were developed to enable both local and remote monitoring of the cultivation processes within the indoor environment.

4.1 Greenhouse Structure

Initially, the greenhouse architecture was modeled in a Three-Dimensional (3D) project in Tinkercad to visualize and understand the system as a whole. At this stage, the arrangements of each component were modeled, as well as the structural dimensions, so that the system's functionality is efficient during cultivation. Figure 4.1 illustrates the physical architecture project.

In this project, the support structure was procured as a module, consisting of a tray measuring 60 cm in length, 60 cm in width, and 12 cm in height, along with a perforated board measuring 56 cm in length, 28 cm in width, and 3 cm in height. This board is designed to drain excess water to deeper layers, as shown in Figure 4.2. The movable cover was constructed from cuts of transparent acrylic, with dimensions of 56 cm in length, 28 cm in width, and 21 cm in height. This cover is assembled and sealed using adhesive materials. As shown in Figure 4.2 illustrates the completed cover placed over

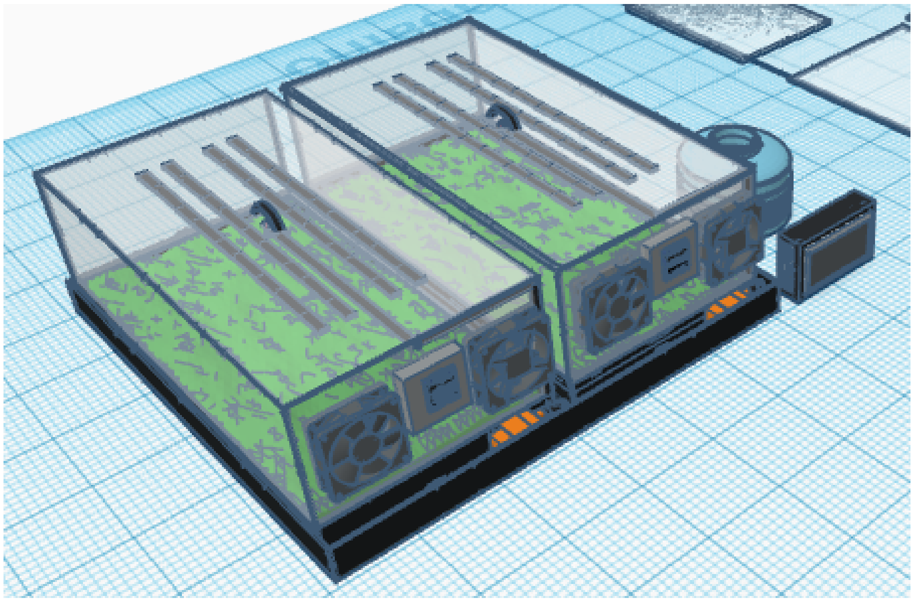


Figure 4.1: 3D design of greenhouse architecture.

the structure.

The irrigation methods were implemented using micro-sprinklers, which cover a larger irrigated surface area, making them ideal for cultivating microgreens in limited spaces [3]. The micro-sprinkler systems comprise a network of components extending from the water reservoir for distribution across the greenhouse covering.

As illustrated in Figure 4.3, the reservoir is a transparent plastic container with a capacity of eight Liters (L). Water is pumped into the tubing from an aquarium micro-pump located at the bottom of the reservoir and powered by 8 Volts (V). The voltage and placement of the pump within the reservoir are crucial to ensure adequate water pressure and uniform distribution across the surface. Water flows through the conduits and dissipates at the emission points.

The emitters are arranged in two rows with four regulated outlets each for gentle dispersion, ensuring no pressure drops during system operation. Figure 4.4 depicts the system composed of hoses, selectors, and emitters located in the upper structure of the greenhouse among the LEDs.

Additionally, atomizers were employed in the center of the greenhouse to complement

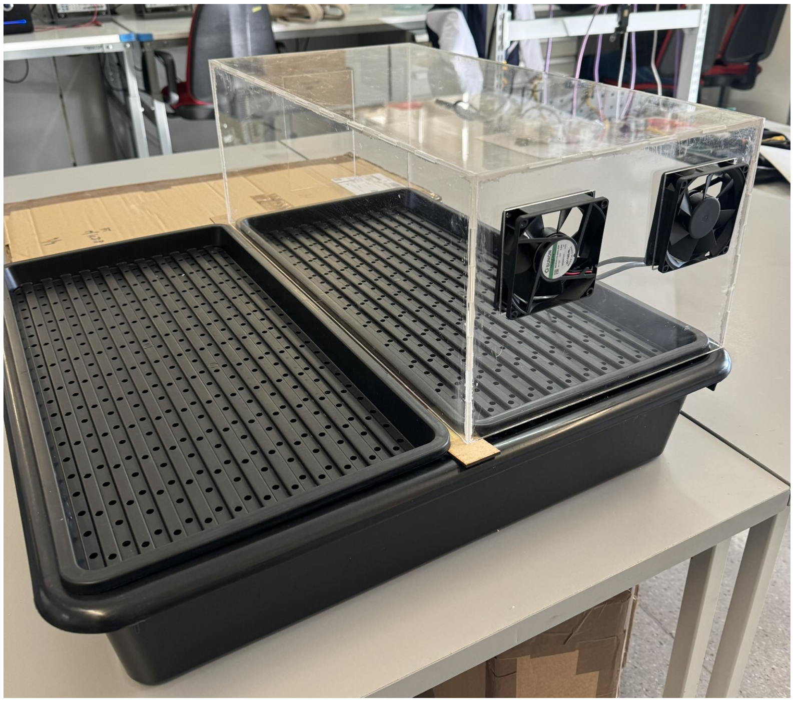


Figure 4.2: Base of the structure consisting of a tray and a perforated board, covered with an acrylic shield.

the irrigation systems, as depicted in Figure 4.5. They disperse water into a fine mist of particles, gently moistening the plants and thereby evenly increasing air humidity distribution [3].

The ventilation mechanisms in the greenhouse facilitate climate control using fans. Two alternating fans were installed on the side structure of the greenhouse to maintain constant air exchange between the indoor and outdoor environments, as depicted in Figure 4.4. While one fan is responsible for exhausting air from inside to outside, the other fan introduces external air into the greenhouse [26].

Additionally, indoor environments feature heaters to complement temperature regulation efforts. These actuators prevent exponential drops in temperature, maintaining it within desirable limits [26].

In this project, two heating systems were installed, each equipped with thermal pads, heat sinks, and fans, as shown in Figure 4.5, so that the heated air is evenly distributed throughout the room. On the side structure, facing the fans, the DHT22 sensor was

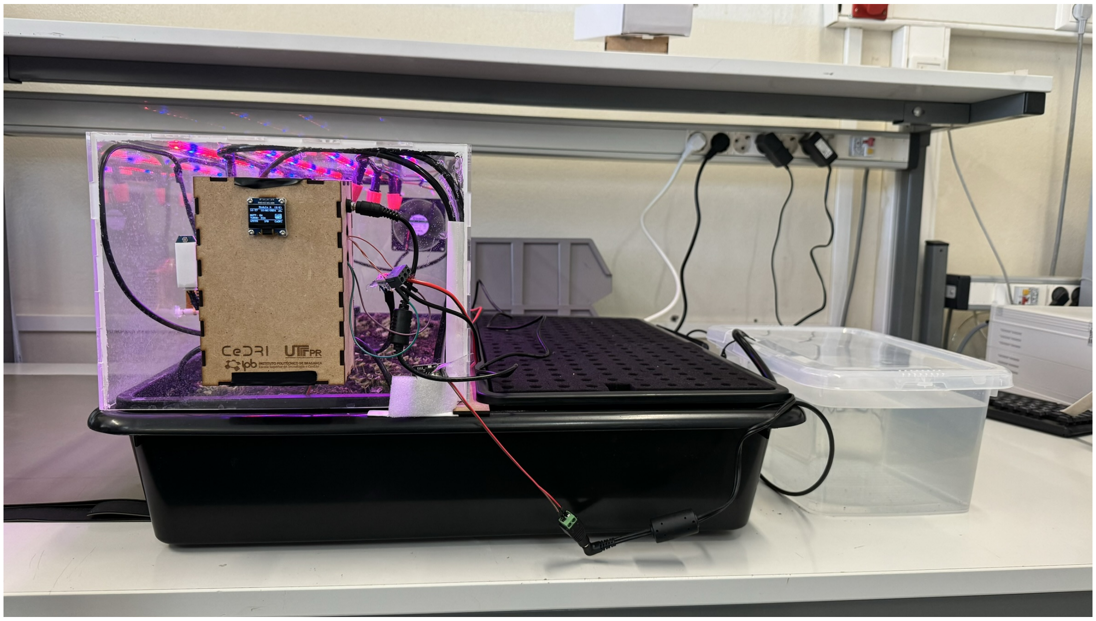


Figure 4.3: Front structure of the greenhouse.

installed to monitor temperature conditions as well as air humidity.

Controlled environments require artificial lighting to simulate ideal sunlight conditions [6]. For this reason, four rows of LED strips with a combination of red and blue lights were distributed on the upper surface of the greenhouse, as illustrated in Figure 4.4. Additionally, the BH1750FVI light intensity sensor was also installed to assist in monitoring.

Finally, the system incorporates technologies to control all devices in the environment through a central circuit. The circuit board is located on the front panel of the greenhouse, with its dimensions designed to fit the circuit's size. Front cutouts were made for the local display, rear cutouts for routing electrical wiring into the greenhouse, and side cutouts for accessing the power supplies.

All sensors and actuators positioned inside the indoor environment are interconnected in this circuit and have been affixed in place using instant adhesives, hot glues, or double-sided tapes. Additionally, an ESP-CAM, based on ESP32 technology, was installed to integrate a compact camera and capture real-time images and videos to enhance monitoring.

The control of the protected environment involves the use of sensors and actuators

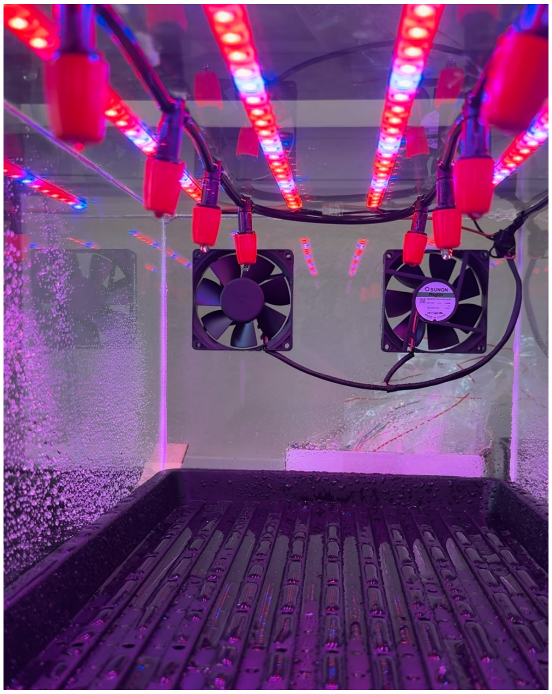


Figure 4.4: Composition of actuators installed in the structure.

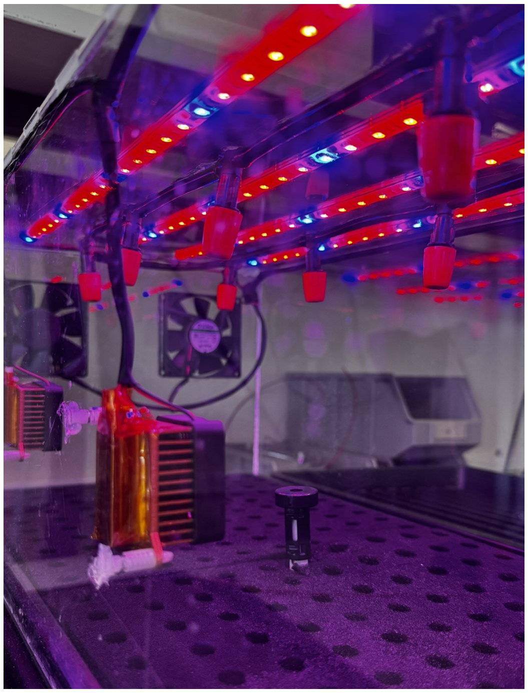


Figure 4.5: Atomization system installed.

that automatically monitor and simulate environmental conditions. This project was composed of modular and independent systems, with controls adaptable to each crop. Thus, it is possible to add new modules to this system by integrating components. The devices integrated into the circuit for monitoring are illustrated in Figures 4.4 and 4.5.

4.2 Remote Interface Device

The algorithms for developing the graphical interface were developed in Node-RED as it offers the necessary tools for monitoring and controlling all parameters. As the objective was to develop an interactive and remote interface, this software had the ideal characteristics for development.

In this work, the indoor environment will be customized according to the culture defined for each crop. To achieve this, different microgreen crops were defined in a menu using a *dropdown* for selection between chard, watercress, lettuce, chicory, chives, carrots, coriander, leafy cabbage, spinach, mint, basil, arugula, and salsas, numbered from 2 to 14. Then, this selection will be stored in a global variable for later considerations as all parameterizations will be customized according to this information.

Depending on the culture, the parameters allow for optimal conditions to be established in the indoor environment during the development of the crop. The optimum temperature factors during the day were established according to the average between their optimum limits, the majority being 20°C, while at night 15°C was considered.

Air and soil humidity factors vary from around 55 % to 80 %, which were considered according to the needs of each crop to keep the soil little or moderately moist [12][13]. Both present optimal factors based on the average of their limits, whose limit conditions are the same during the day and night. Finally, the luminosity was considered standard during the day at 100 % and during the night at 0 % to simulate the length of the day. The Table 4.1 locates the optimal conditions considered for each microgreen.

Microgreen	**Temp$_{\text{day}(°C)}$**	**Temp$_{\text{night}(°C)}$**	**Hum.Air$_{\text{day}(\%)}$**	**Hum.Air$_{\text{night}(\%)}$**	**Hum.Soil$_{\text{day}(\%)}$**	**Hum.Soil$_{\text{night}(\%)}$**	**Lux$_{\text{day}(\%)}$**	**Lux$_{\text{night}(\%)}$**
Chard	20	15	65	65	60	60	100	0
Cress	20	15	60	60	65	65	100	0
Lettuces	17.5	15	65	65	70	70	100	0
Almeirão	20	15	65	65	65	65	100	0
Chives	20	15	60	60	70	70	100	0
Carrot	18	15	70	70	70	70	100	0
Coriander	20	15	60	60	60	60	100	0
Leafy Cabbages	20	15	65	65	70	70	100	0
Spinach	17.5	15	70	70	70	70	100	0
Mints	20	15	60	60	65	65	100	0
Basil	20	15	75	75	75	75	100	0
Arugula	20	15	60	60	65	65	100	0
Salsas	20	15	60	60	60	60	100	0

Table 4.1: Optimal conditions for microgreens [12][13].

The values of the ideal parameters are alternated according to the alternation of shifts received by MQTT. If the shift is true, it is considered the daytime period, if not, the nighttime period is considered.

All these parameters are regulated according to the specifications of each culture and sent to the control system via MQTT. Furthermore, variables are set and displayed in the visual interface for monitoring. Figure 4.6 represents the combination of nodes referring to the data flow to define the variables for control according to the microgreen selection.

Furthermore, the optimal limit conditions for each crop were stored as objects and will

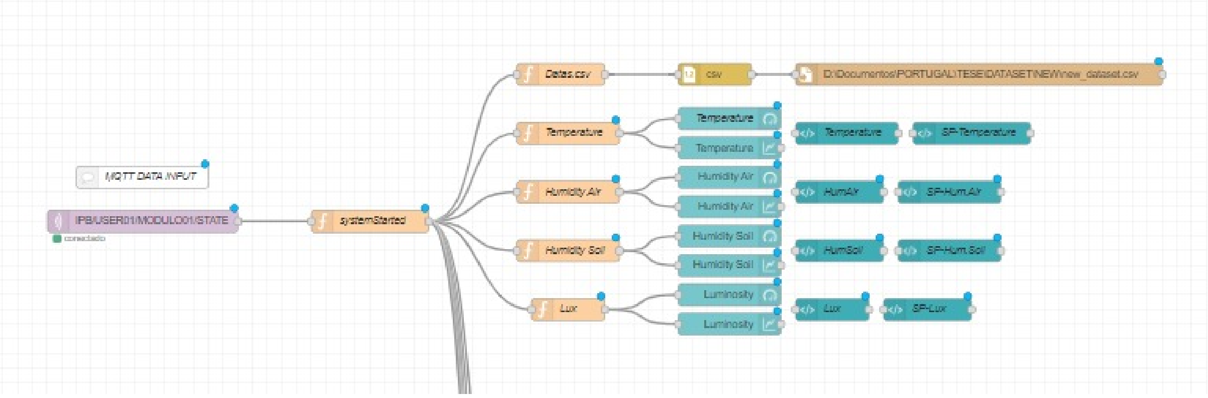


Figure 4.6: Flow between nodes to establish revenues.

be processed by control structures *switch case* according to the selected crop, establishing its limits for the best crop yield.

Therefore, temperature limits were considered to be 15 °C to 25 °C during the day for almost all microgreens except lettuce, carrots, and spinach. On the other hand, during the night, as temperatures naturally drop in the environment, limits between 10 °C to 15 °C were considered, except for the crops mentioned [12][13].

The minimum limits of air humidity were considered as 55 % for almost all crops, while soil humidity limits were considered to be between 50 % to 80 % [12][13]. These limit conditions for each culture, during the day and night, are represented in Tables 4.2 and 4.3, respectively.

Microgreen	**Temp$_{max(°C)}$**	**Temp$_{min(°C)}$**	**Hum.Air$_{min(\%)}$**	**Hum.Soil$_{max(\%)}$**	**Hum.Soil$_{min(\%)}$**	**Lux$_{(\%)}$**
Chard	15	25	65	50	70	100
Cress	15	25	60	60	70	100
Lettuces	15	20	65	60	80	100
Almeirão	15	25	65	60	70	100
Chives	15	25	60	60	80	100
Carrot	15	21	70	60	80	100
Coriander	15	25	60	50	75	100
Leafy Cabbages	15	25	65	60	80	100
Spinach	15	20	70	60	80	100
Mints	15	25	60	60	70	100
Basil	15	25	50	70	80	100
Arugula	15	25	60	60	70	100
Salsas	18	22	55	50	70	100

Table 4.2: Daytime boundary conditions for microgreens [12][13].

All parameters monitored by the sensors must be within the stipulated optimal limits

Microgreen	**Temp**$_{\mathbf{max(^\circ C)}}$	**Temp**$_{\mathbf{min(^\circ C)}}$	**Hum.Air**$_{\mathbf{min(\%)}}$	**Hum.Soil**$_{\mathbf{max(\%)}}$	**Hum.Soil**$_{\mathbf{min(\%)}}$	**Lux**$_{\mathbf{(\%)}}$
Chard	10	15	65	50	70	0
Cress	10	15	60	60	70	0
Lettuces	10	15	65	60	80	0
Almeirão	10	15	65	60	70	0
Chives	10	15	60	60	80	0
Carrot	10	15	70	60	80	0
Coriander	10	15	60	50	75	0
Leafy Cabbages	10	15	65	60	80	0
Spinach	10	15	70	60	80	0
Mints	10	15	60	60	70	0
Basil	10	15	50	70	80	0
Arugula	10	15	60	60	70	0
Salsas	13	18	55	50	70	0

Table 4.3: Night boundary conditions for microgreens [12][13].

so that the microgreen's behavior is as expected. The monitoring system continuously checks these parameters and issues an alert whenever the recorded values are below the minimum limits or above the maximum limits. Figure 4.7 outlines the flow between nodes for identifying the culture and issuing alerts.

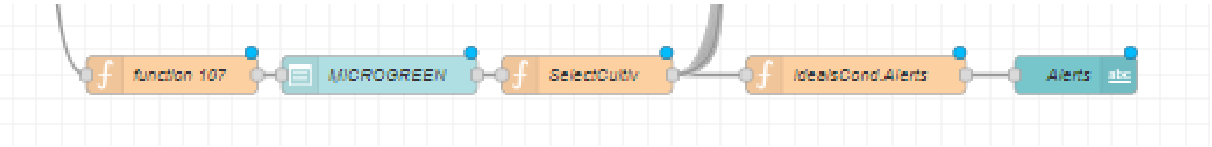


Figure 4.7: Flow between nodes for alerts depending on crop types and ideal conditions.

The system comprises automatic and manual operating modes for controlling devices such as valves, atomizers, lighting, heaters, and ventilation. When the system is in manual mode, the user can remotely control the actuators located in the greenhouse through the graphical interface.

For this reason, sliders were implemented by *sliders* to adjust the intensity of the actuators and *switches* acting as switches to activate or deactivate certain functions. Furthermore, cultivation will be counted when *switch play* is activated and stored in a global variable.

On the other hand, when automatic mode is enabled, the greenhouse's internal controls will be automated according to the conditions established according to the crop. Furthermore, the controls displayed in the graphical interface will be disabled to prevent any modifications to the system, configured using the *enabled* condition in the controls and setpoints. Figure 4.8 shows the combination of nodes for the control flow sent to MQTT.

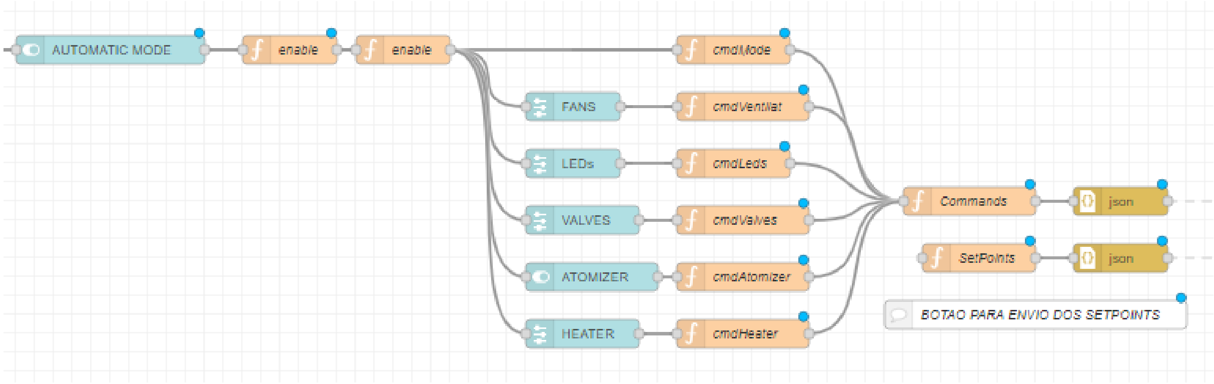


Figure 4.8: Control flow sent to MQTT.

To monitor the conditions of the indoor environment, combinations between nodes were used to receive data from sensors and actuators continuously, being displayed in an understandable way in the graphical interface. Data acquisition from devices is received through the connection to the MQTT broker by subscribing to a specific topic, where data from sensors and actuators are being published.

Then, the data is recognized and forwarded directionally to the graphical interface for temperature, air humidity, soil humidity, and luminosity. The graphics were configured according to the visualization established for historical and real-time data to facilitate monitoring of the greenhouse's internal conditions.

The combination scheme between the nodes, represented in Figure 4.9, locates the data flow from the established sensors, being distributed to each variable individually. The functions that precede the dashboards select specific information within a set of data read from MQTT.

On the other hand, actuator data is directed to displaying operating time, as each message received includes information about the actuator status, such as active or inactive. Operation time is calculated from a global variable indicating whether cultivation has started.

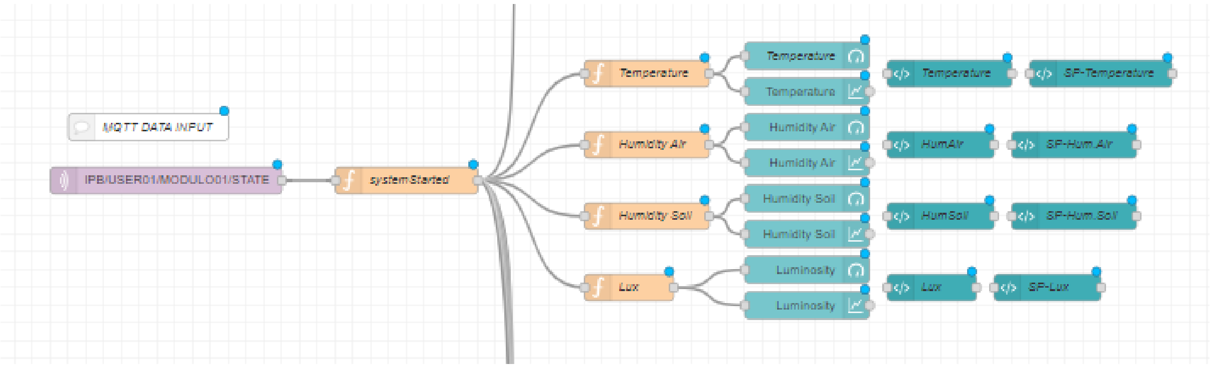


Figure 4.9: Flow of sensor data sent by MQTT.

If cultivation has started, the status of each actuator is checked to start counting the operating time. This count is processed only when the state is active, stopped when it is inactive, and resumes when it is active again. The variables of each actuator are updated and increased in the operating time accumulated during the cultivation cycle. The schematic can be seen in Figure 4.10.

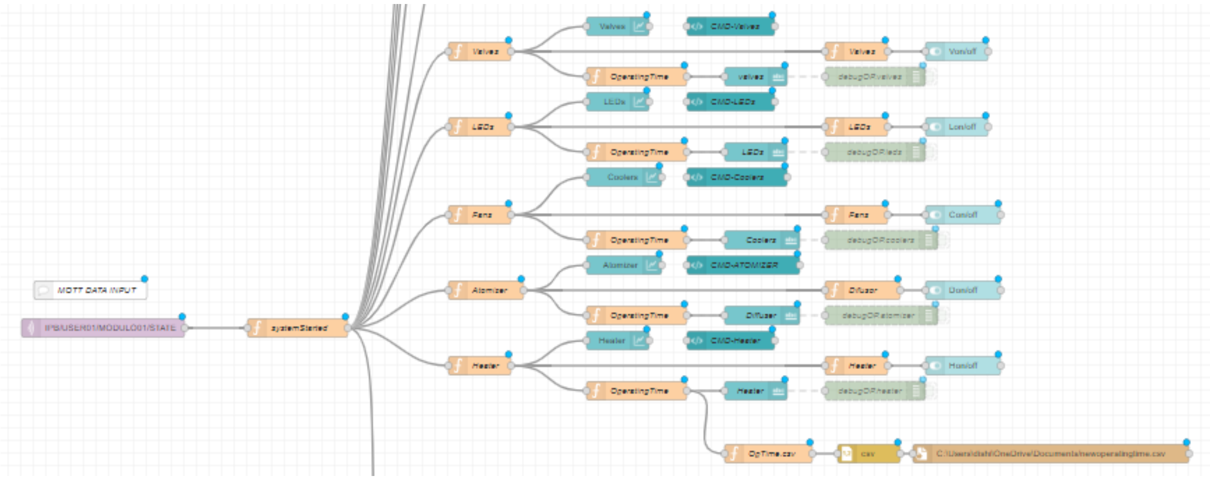


Figure 4.10: Flow of actuator states sent by MQTT.

The dashboards include graphical visualization of actuator states over time in a line graph and sampling of operating time formatted in days, hours, minutes, and seconds. Additionally, the current state is also displayed by a *switch* being toggled between operating modes. They are configured with the *flash on* icon in green when it is on and *flash off* in gray when it is off.

Real-time monitoring also includes video monitoring with internal images of the greenhouse from a local camera. The video stream was incorporated into the dashboard through a template configured to access images available on a specific HyperText Markup Language (HTML) page, where the images are being published. Access is restricted to the local network where the camera is installed.

These dashboards were organized into group widgets defined according to specific functional areas, interactive elements, and visual style to facilitate understanding and operation during monitoring. The structure contains sizes and styles of widgets of varying shapes, configured internally as illustrated in Figure 4.11.

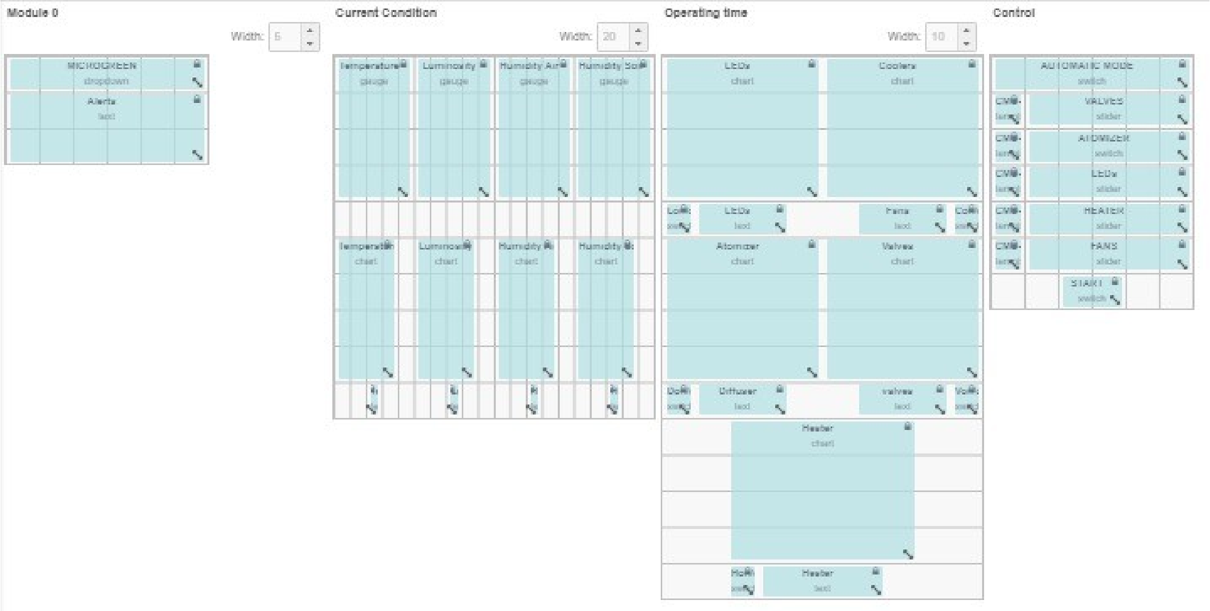


Figure 4.11: Structuring some dashboards in the graphical interface.

Additionally, icons were added via *templates* to the dashboards to identify functionalities and keep monitoring more intuitive in the graphical interface. The icons were created from *Material Icons*, which includes a wide choice of icons for user interfaces that are easily recognized by the program.

All data models JavaScript Object Notation (JSON) are sent to a file Comma-Separated Values (CSV)so that all conditions are stored during cultivation. Three databases specified for the general data set, forecast data, and operating time data, structured in JSON formats, were created in the My Structured Query Languag (MySQL) database for data reading and writing operations.

The data set was structured in a FIWARE model in order to help standardize information exchanges between devices. The Figures 4.12 and 4.13 represent the standardization used in the data models so that devices at the edge layer receive command and setpoints information, respectively.

```
var payload = {
    id: "Node-red-USER-NOME-PC",
    type: "APP-Node-red-ControlGreenhouse",
    address: {
        addressLocality: "Bragança",
        addressCountry: "PT"
    },
    dataProvider: "node-red",
    dateObserved: "time",
    hasAgriParcelParent: "GreenhouseModule",
    operationMode: comando1,
    cmdActuators: {
        ventilationFan: comando2,
        ledStrip: comando3,
        waterValve: comando4,
        atomizer: comando5,
        heatingPad: comando6,
        startMicrogreen: comando7
    }
};

// Converte o objeto JSON em uma string
msg.payload = JSON.stringify(payload);
```

Figure 4.12: Structuring the database for commands in a Fiware model.

```
var payload = {
    id: "Node-red-USER-NOME-PC",
    type: "APP-Node-red-ControlGreenhouse",
    address: {
        addressLocality: "Bragança",
        addressCountry: "PT"
    },
    dataProvider: "node-red",
    dateObserved: "time",
    hasAgriParcelParent: "GreenhouseModule",
    morningSetpoint: {
        idealTemperature: setPoint1,
        idealSoilMoisture: setPoint2,
        idealHumidity: setPoint3,
        luminosityTime: setPoint4,
        luminosityLux: setPoint5
    },
    nightSetpoint: {
        idealTemperature: setPoint6,
        idealSoilMoisture: setPoint7,
        idealHumidity: setPoint8,
        luminosityTime: setPoint9,
        luminosityLux: setPoint10
    },
};

// Converte o objeto JSON em uma string
msg.payload = JSON.stringify(payload);
```

Figure 4.13: Structuring the database for setpoints using a Fiware model.

The files were created by specifying the directory and all columns according to the names predefined in the function that receives the data, separated by a comma without spacing, and configured so that the first line is composed of these names. Figure 4.10 illustrates the flow for creating datasets from the *write file* node as the last flow located in the schematic.

Furthermore, the system must resume its operations from the state in which it stopped, ensuring that no data is lost and that automation is continuous throughout the cultivation. For this reason, functions were implemented to check the program initialization to reestablish the current state based on database readings, as shown in Figure 4.16.

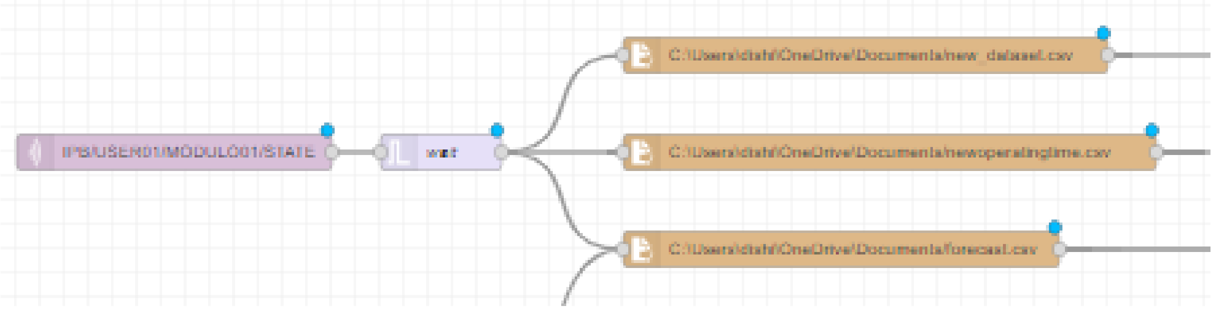


Figure 4.14: Flow for reading data sets.

Finally, as access to the interface is open to the IP address, security systems were configured to authenticate the Node-RED Dashboard. This configuration was implemented in the *settings.js* file located in the Node-RED installation folder. Using authentication protects access to program files and interactive interfaces to authorized users only.

The system includes local or remote visual interfaces for monitoring crop parameters located indoors. The Organic Light-Emitting Diode (OLED) display was integrated into the control module, allowing local visualization of fundamental information, such as temperature, air and soil humidity, luminosity, mode, actuators, setpoints, current date, shift, connection to WiFi, and MQTT. All information is distributed across six microscreens alternating every few seconds.

Furthermore, the Node-RED graphical interface also supports remote monitoring on a Raspberry Pi or any device connected to the network. The system presents real-time data, remote controls, and historical graphs through a web page or mobile application.

4.3 Cycle Change Prediction

The system includes a mathematical crop prediction model to monitor the growth of microgreens. The algorithms employed calculate the time remaining until the end of cultivation of different crops using the concept of GDD accumulated every hour, as shown in Figure 4.15. However, this concept only applies when all internal factors are in optimal conditions under the requirements of each crop.

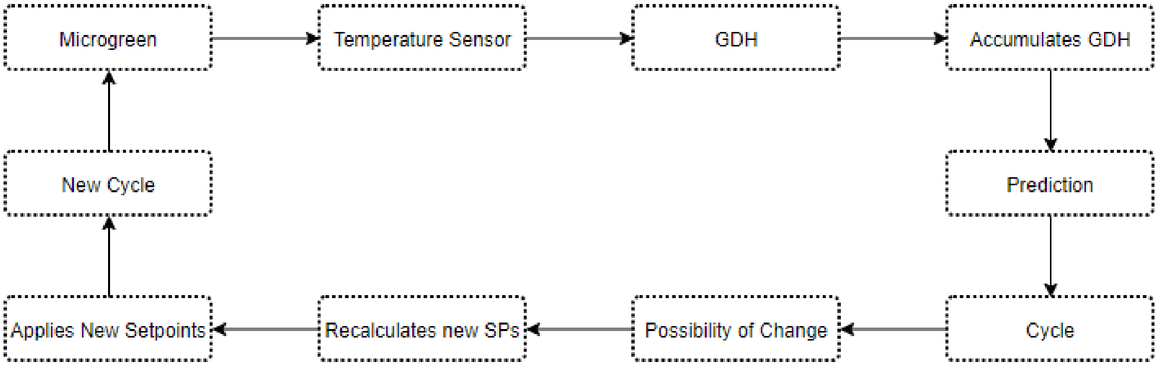


Figure 4.15: Structuring for forecast calculations and the possibility of changing the cycle.

Each microgreen has a specific maturation time that varies according to the species and can be influenced by environmental factors. Microgreens are harvested, on average, between 7 and 21 days after germination, representing only 36 % of the complete vegetable cycle [2][6][7].

For this reason, specific conditions were considered regarding the maturation time of each microgreen, being 36 % over the complete cycle of each crop, disregarding the germination time. Then, the germination time was added to complement the complete cultivation time of the microgreens, as shown in the equation (4.1).

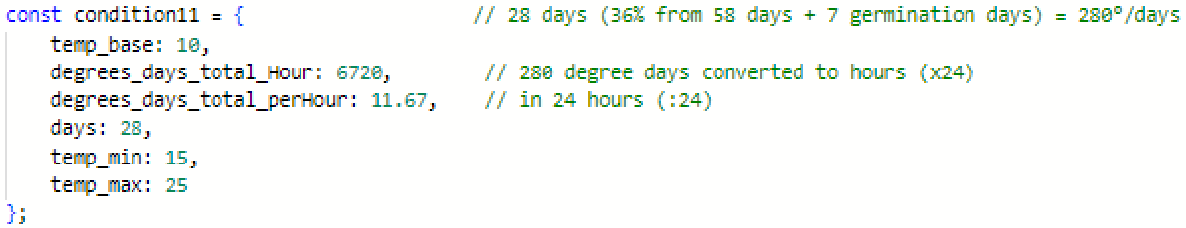
$$\text{Cycle}_{\text{microgreen}} = (36\ \% \times (\text{Cycle}_{\text{total}} - \text{Days}_{\text{germination}})) + \text{Days}_{\text{germination}} \tag{4.1}$$

Thermal accumulation in GDD comprises the sum of daily temperatures above a base temperature necessary for crop growth. To do this, it is necessary to convert the complete cycle of each microgreen in days to Degree-Days (°C/days), as shown in the equation (4.2).

$$\mathrm{GDD_{total}} = (\mathrm{T_{dailyAverage}} - \mathrm{T_{base}}) \times \mathrm{Days_{totals}} \tag{4.2}$$

where $T_{\text{dailyAverage}}$ is the average daily temperature, T_{base} is the base temperature of a given crop and $Dias_{\text{totals}}$ are the total days until the end of cultivation. In this context, it is possible to measure the degree-days necessary for the complete crop cycle.

Then, the GDD values (°C/days) were converted into Growing Degree Hours (GDH) (°C/ hours) to consider the maximum and minimum temperatures every hour and increase the precision in the calculations. In this way, the total GDHs of each crop were calculated by multiplying the total GDDs by 24 hours. Figure 4.16 represents one of the conditions considered, for example for the cultivation of basil, which were also applied to other crops.



```
const condition11 = {                    // 28 days (36% from 58 days + 7 germination days) = 280°/days
    temp_base: 10,
    degrees_days_total_Hour: 6720,        // 280 degree days converted to hours (x24)
    degrees_days_total_perHour: 11.67,    // in 24 hours (:24)
    days: 28,
    temp_min: 15,
    temp_max: 25
};
```

Figure 4.16: Basil maturation conditions as a microgreen.

With this, the values for the prediction calculations of each crop were obtained, located in Table 4.4.

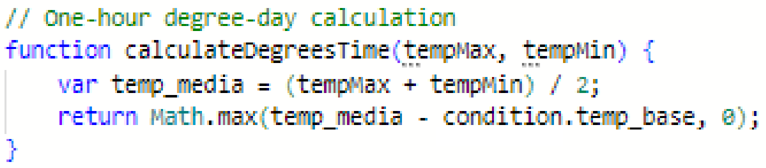
Initially, each time the program is updated, it checks whether the cultivation has started and which microgreen will be cultivated based on global variables. If cultivation has started, the code processes historical data from a file, separating the data into rows and columns. Then, the maximum and minimum temperatures are obtained every hour and processed by the formula represented in the line of code located in Figure 4.17.

where T_{max} is the maximum hourly temperature, $T_{\text{min}}$is the minimum hourly temperature and T_{base} is the base temperature of a given crop.

GDHs will be accumulated and stored every hour while the crop is developing. The

Microverde	**Days$_{\text{totals}}$**	**Hours$_{\text{totals}}$**	**GDD$_{(°\text{C/days})}$**	**GDH$_{(°\text{C/hours})}$**	**Temp$_{\text{base}(°\text{C})}$**
Chard	25	8	250	6000	10
Cress	28	6	280	6720	10
Lettuces	25	5	186	4464	10
Almeirão	29	9	290	6960	10
Chives	40	11	400	10560	10
Carrot	51	18	408	9792	10
Coriander	45	11	450	10800	10
Leafy Cabbages	36	8	360	8640	10
Spinach	23	7	172.5	4140	10
Mints	45	11	450	10800	10
Basil	28	7	146	3495	10
Arugula	32	7	320	7680	10
Salsas	52	30	520	12480	10

Table 4.4: Microgreen harvest periods [12][13].



```
// One-hour degree-day calculation
function calculateDegreesTime(tempMax, tempMin) {
    var temp_media = (tempMax + tempMin) / 2;
    return Math.max(temp_media - condition.temp_base, 0);
}
```

Figure 4.17: Calculation of GDH every hour.

remaining days will be calculated by the difference between total GDHs and GDHs accumulated to date, as shown in the equation (4.3). Additionally, the remaining GDHs will also be calculated by converting the remaining days and stored in a global variable for later applications.

$$\text{Days}_{\text{remaining}} = \text{Days}_{\text{totals}} - \left(\frac{\text{Days}_{\text{totals}} \times \text{GDH}_{\text{accumulated}}}{\text{GDH}_{\text{totals}}} \right) \tag{4.3}$$

where $GDH_{\text{accumulated}}$ are the Degree-Hours accumulated to date and GDH_{totals} are the total Degree-Hours for the end of cultivation.

Finally, the algorithm recognizes the database update from the last *timestamp* considered and receives the new data to be processed in the formula, accumulated, and updated in the remaining days. When the remaining days are less than or equal to zero, counting stops and the remaining days are equal to zero.

The schematic flows illustrated in Figure 4.18 comprise the system presented on crop forecasting, displaying graphics and texts that demonstrate the days remaining for harvests. Furthermore, data is stored in a database to ensure system continuity.

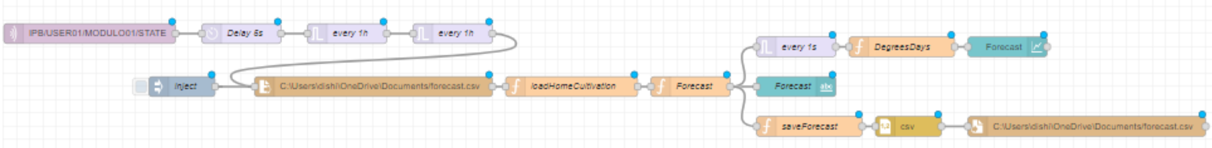


Figure 4.18: Schematic flow for crop forecasts.

The indoor environment can also be configured according to the cycle desired by the producer. According to harvest forecast estimates, the producer will be able to adjust the cultivation system according to their needs. To do this, the system includes a menu from a *dropdown*, illustrated in Figure 4.19, to select whether the crop will be developed under normal conditions, advanced conditions, or delayed conditions, numbered from 1 to 3, respectively.

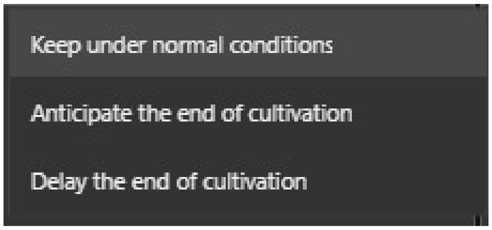


Figure 4.19: Different crop cycle options.

If the cycle is 1, only information is displayed that the crop is in normal conditions. If the cycle is 2, the algorithm checks the possibility of bringing the crop forward and calculates how many days can be brought forward considering the maximum temperature in all circumstances. If the cycle is 3, the algorithm checks the possibility of delaying the cultivation and calculates how many days can be delayed assuming the minimum temperature in all conditions.

If cultivation is advanced or delayed, the user can enter the desired value between the pre-established days for change. If the user enters higher numbers or unknown characters, an alert will be generated that the entered value is invalid or non-numeric. Therefore, the user must enter another value again and press the button *switch* to confirm the change.

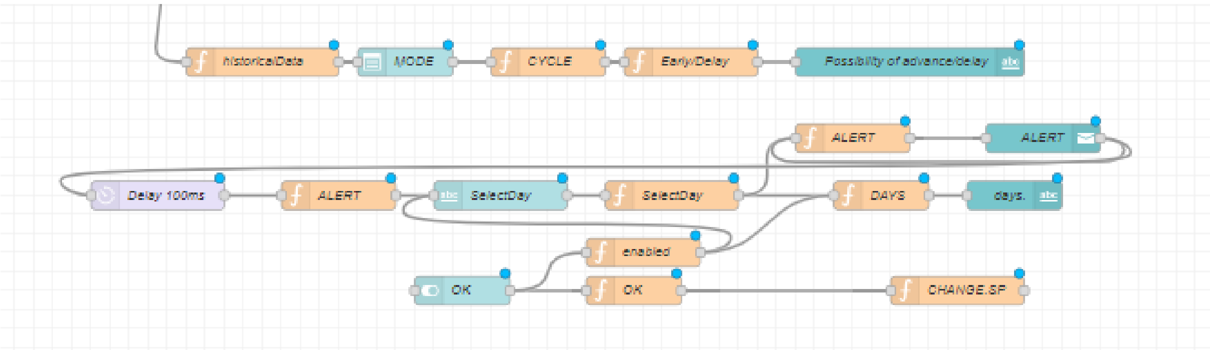
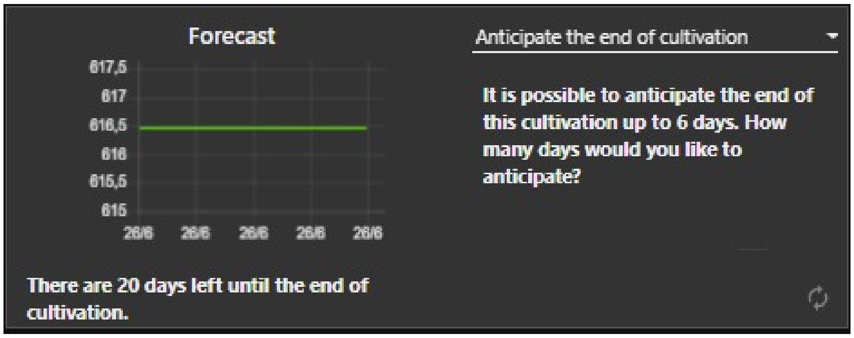


Figure 4.20: Schematic flow for combination of all conditions.

When the change is established, the algorithm will receive the days for adjustment and must adjust the temperature according to the defined cycle. If the option is to advance the cycle, the code reduces the remaining days, calculates the corresponding hours and redistributes the sum of the remaining GDHs in the hours remaining until the end of the cultivation. With this, a new temperature will be updated to reach the remaining GDHs on the established days and will be sent to the setpoints to update the recipe.



There are 20 days left until the end of cultivation.

It is possible to anticipate the end of this cultivation up to 6 days. How many days would you like to anticipate?

Figure 4.21: Crop forecast model located in the graphical interface.

On the other hand, if the option is to delay the cycle, the code extends the remaining days and redoes the calculations to determine the new temperature for the recipe, as in the previous condition. The figures 4.20 e 4.21 present the schematic for the combination of all conditions and the model displayed in the graphical interface, respectively.

According to Figure 4.20, the flows combine the nodes in order to establish the verification of the established cycles and the days entered by the user so that the algorithm can redo the calculations and change the setpoints according to the defined requirements.

Chapter 5

Tests and Results

This chapter discusses the tests conducted and the results obtained. Initially, vacuum tests were conducted to validate the functionality of the systems. Subsequently, the results of the system were verified during basil cultivation, monitoring performance under the established conditions.

The tests were designed to verify the overall functionality of the structure, the connections between components, the capability for real-time monitoring, the integrity of the historical database, and the quality of microgreen cultivation.

5.1 Empty Tests

These systems combine sensors and actuators for regular factors such as temperature, humidity, illumination, and illumination. To ensure the effectiveness of the system as a whole, isolated tests were carried out to verify the functionality of each component, and then they were integrated into the system, and new functionality tests were developed on the system as a whole. Therefore, the components tested were micro sprinklers, atomizers, heaters, fans, LEDs, and sensors.

Irrigation Tests

The irrigation system required several tests to approximately adjust a continuous distribution. First, we check the functionality of a set of elements for creating the irrigation system, as illustrated in Figure 5.1. Irrigation systems were supplied from a water reservoir, where water was pumped through ducts and distributed by micro sprinklers. However, the first test was carried out with a low-voltage water pump and did not go as expected, as the irrigation was concentrated in a single point instead of being distributed across the surface.

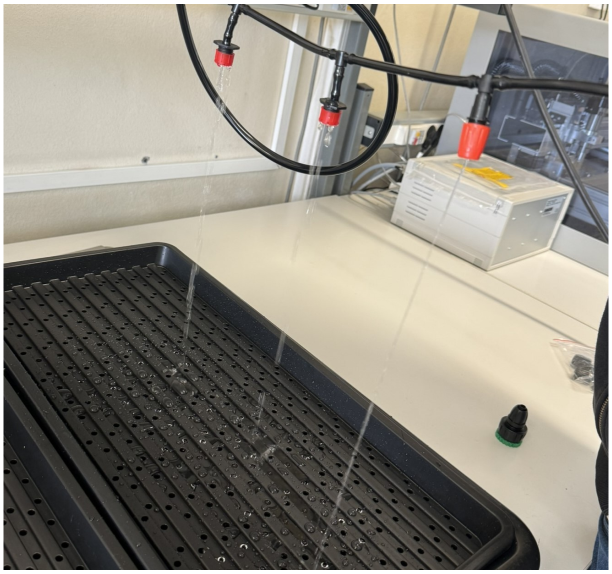


Figure 5.1: First irrigation test.

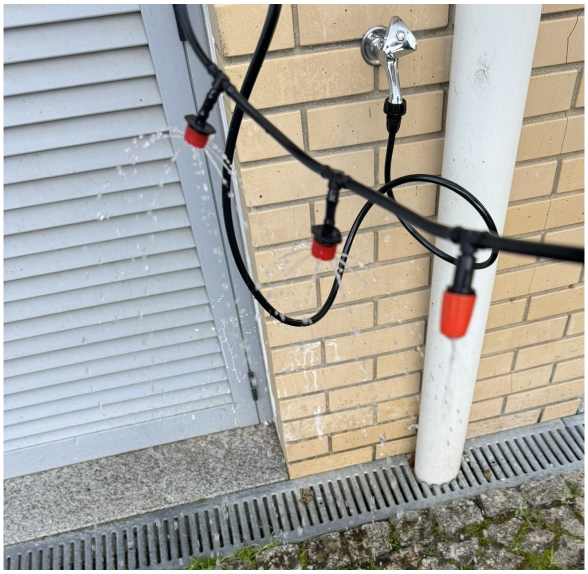


Figure 5.2: Second irrigation test.

This test was repeated under the same circumstances with water flowing directly from a tap, as shown in Figure 5.2. The result was as expected and problems with water pressure were considered. As a result, the components and distributions were reformulated in order to increase the water pressure in the irrigation systems. In this context, a 8 V voltage was configured to power the water pump, which was glued with instant glue to the bottom of the reservoir. Both configurations were used to increase the water pressure on the pump.

The system distributed in this way increased the water pressure in the irrigations and the micro sprinklers performed as expected. Initially, six emitters were considered equally distributed over two rows on the upper face of the greenhouse. However, to provide greater surface coverage, a new emitter was added to each row without reducing the water pressure.

With the irrigation system installed, tests were carried out with plastic cups distributed across the tray to check whether the water distribution would be uniform across the surface, as shown in Figure 5.3. According to Figure 5.4 it was possible to verify the distribution concentrated in some points in the central area and in the corners of the structure with little water inside the cups.

Figure 5.3: Testes de distribuição de água.

Figure 5.4: Resultados da distribuição de água.

For this reason, standard substrates were used to distribute continuous water across the surface from irrigation. Furthermore, these systems include micro-sprinklers adjusted to an opening that distributes water in microparticles across the surface without concentrated water drops that could damage the crop.

To complement systems for relative humidity, an atomizer was installed that absorbs water from the soil and vaporizes it into the environment, increasing air humidity inside the greenhouse, illustrated in Figure 5.6. Operational tests were carried out on this actuator and it was verified that the filter needs to have absorbed a good amount of water, be well fixed to the bottom and have direct contact with the steam outlet to function properly. Therefore, a piece of phenolic foam was placed at the base of the filter to fix the atomizer to the substrate and aid water absorption.

Fan and Heater Tests

As the cultivation system depends considerably on temperature adjustments to control planting cycles, ventilation and heating systems are essential. The heating of the greenhouse comes from installed heaters while the air circulation comes from fans, both installed on the sides of the structure.

Initially, tests were carried out using the fans and, immediately, they had the expected results of air exchange between the external and internal environment of the greenhouse, reducing the internal temperature. Next, it was necessary to implement a heating system to complement temperature control.

In this sense, thermal pads were inserted for internal heating, which were tested individually and their heating was considerable. However, the heating was only in the thermal pad itself, without disseminating it into the environment.

Therefore, the tests were redone, coupling the thermal pads to the fans with the aim of distributing the hot air throughout the greenhouse, as seen in Figure 5.7. The result was not as expected since the fans cooled the thermal pads instead of distributing hot air.

For this reason, new components were invested to complement the heating system. With this, the thermal pads, micro heat sinks, and a micro fan integrated with glues and thermal tapes were coupled, as shown in Figure 5.5. The heating system with the integration of these elements showed rapid heating results and up to, on average, 10 °C above the previous temperature.

Figure 5.5: Heating tests with thermal pads attached to heat sinks.

Fan and Atomizer Tests

The air humidity control tests were carried out by the actuators of the ventilation and atomization systems, monitored by the DHT22 sensor. The atomizers were initially activated, increasing the air humidity, as shown in Figure 5.6. Then, the atomizers were deactivated and the fans were activated, reducing air humidity, as illustrated in Figure 5.7.

Figure 5.6: Air humidity tests using atomizers.

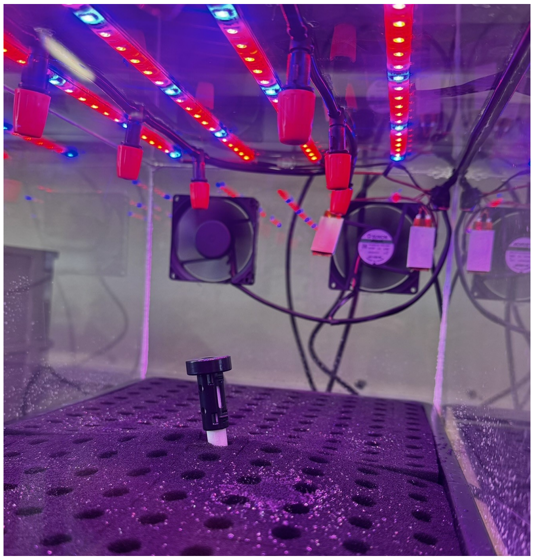


Figure 5.7: Air humidity tests using fans.

The tests worked as expected since these actuators must be enabled at alternate times, as they are inversely proportional. If both actuators are activated simultaneously, operation performance decreases and energy consumption is maintained without causing effects on the environment.

Communication Tests

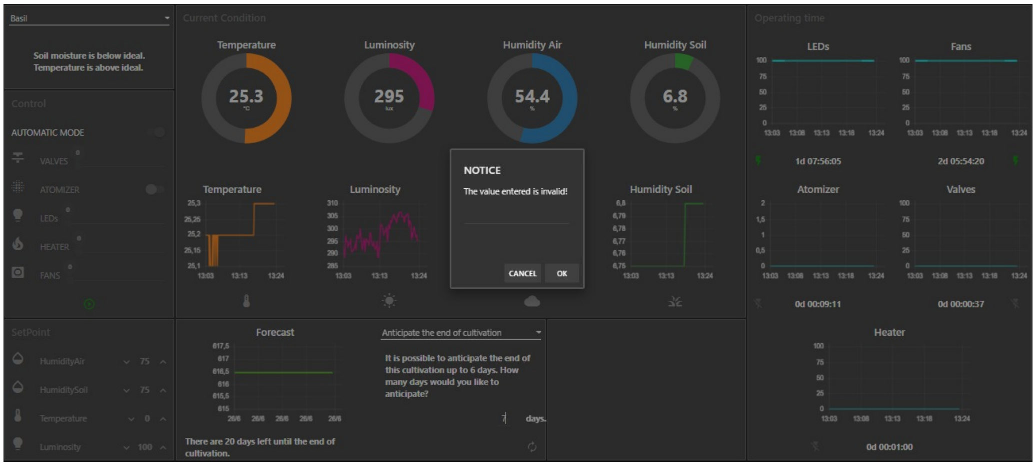
To support the monitoring of all sensors and actuators, communications between Node-RED and the MQTT broker were verified according to specific configurations. Then, tests were carried out using the *debug* node in the MQTT input stream on Node-RED, verifying the publication of the data.

Furthermore, the commands to be sent via MQTT to the control systems were tested by making changes to the dashboards and checking the activation or deactivation of the actuators arranged in the greenhouse. All these tests were monitored in real-time with the help of dashboards in the graphical interface.

Cycle Change Tests by the Producer

According to crop forecasts, the system makes it possible to change the cycle according to the producer's needs. Therefore, the graphical interface allows interaction with the user to enter new cultivation conditions. In this sense, the user changes the cycle and enters the number of days he intends to change according to the possibility of change days imposed by the system.

If the user enters non-corresponding values, the system issues an alert so that the change can be reconsidered again, as illustrated in Figure 5.8. These tests were carried out so that the system recognizes the permitted values and adjusts the necessary parameters, respecting the limits of each culture.



Soil moisture is below ideal.
Temperature is above ideal.

NOTICE

The value entered is invalid!

HumidityAir 75

HumiditySoil 75

Temperature 0

Luminosity 100

It is possible to anticipate the end of this cultivation up to 6 days. How many days would you like to anticipate?

7 days.

There are 20 days left until the end of cultivation.

1d 07:56:05

2d 05:54:20

0d 00:09:11

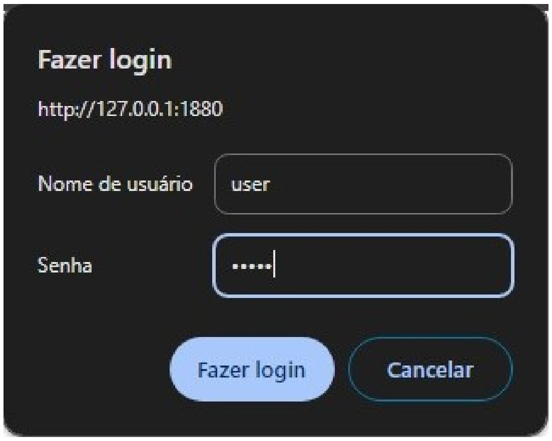
0d 00:00:37

0d 00:01:00

Figure 5.8: Warning when entering invalid values for cycle change.

Authentication Tests

Remote access was tested on different devices, such as cell phones, web pages, Raspberry Pi, and OLED displays, connected via IP address. IP address security systems were also tested, requiring authentication for access and remote interaction with the graphical interface according to the configured security system, as shown in Figure 5.9.



Fazer login

http://127.0.0.1:1880

Nome de usuário: user

Senha: •••••

Figure 5.9: User authentication system.

5.2 Plantation Results

After performing the functional tests and system adjustments, the cultivation began with basil sowing. The seeds were distributed throughout the substrate so that they grew uniformly over the entire surface, as shown in Figure 5.10.

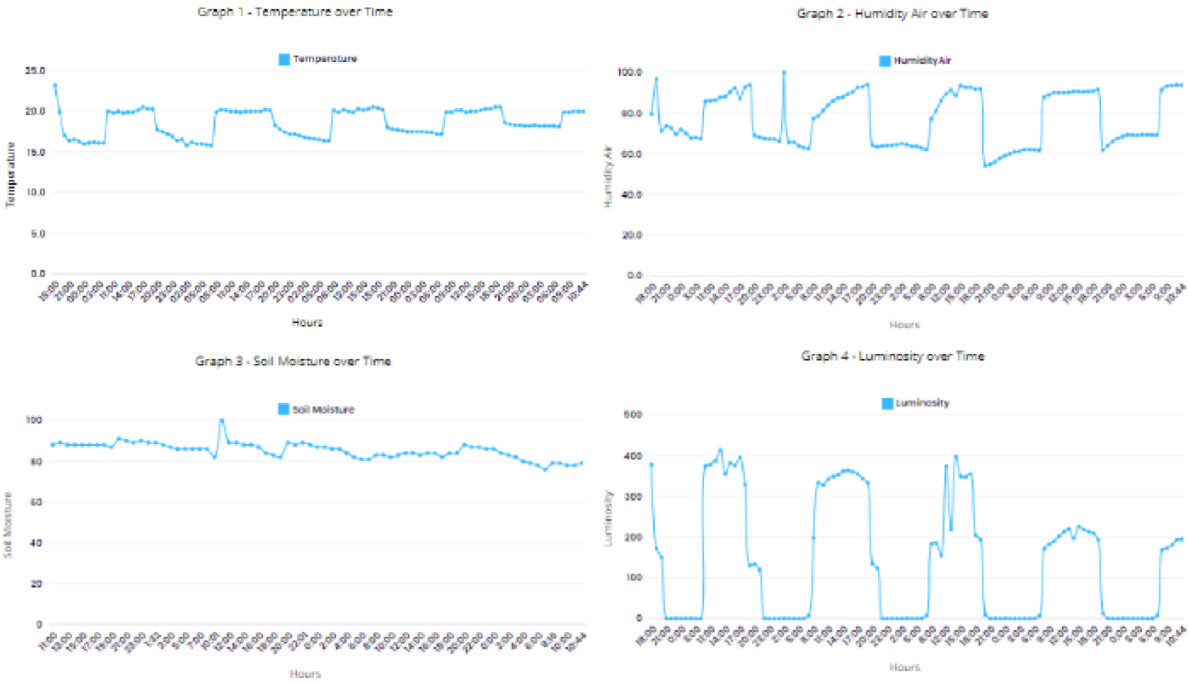
Figure 5.10: Cultivation of seeds in the substrate.

Next, the system was configured according to basil cultivation, establishing the parameters for this species of microgreen. Furthermore, the control systems were configured to cultivate in automatic mode according to established recipes.

The parameters were monitored from germination, which lasted, on average, 7 days until the harvest period, comprising a complete cycle of 28 days. All parameters located in the indoor environment during this period were stored in a database for the analysis of historical data and the recomposition of all of them in the graphical interface.

According to the samples located in the Figure 5.11, temperatures settled approximately between optimum values with small changes. However, on hotter days the temperature showed greater fluctuations without reaching optimal values. In this sense, it was observed that the internal temperature remained limited according to the ambient temperature since the control system only includes ventilation systems between the internal and external environment.

Soil moisture remained between its optimum values, increasing considerably during irrigation, as illustrated in Figure 5.11. While air humidity remained below its optimal


Graph 1 - Temperature over Time

Graph 2 - Humidity Air over Time

Graph 3 - Soil Moisture over Time

Graph 4 - Luminosity over Time

Figure 5.11: Environmental factors as a function of time during cultivation.

values due to little activation of the atomizer, as the fans remained on to control the temperature.

Finally, the luminosity represented the daytime periods at 100 % and nighttime periods at 0 %, being turned on during the day at their maximum capacity and turned off during the night, as shown in Figure 5.11.

All of these parameters were monitored in real-time through the graphical interface, which, in turn, also includes images displayed live to complement microgreen monitoring, as seen in Figure 5.12.

The operating times of each actuator depend on the values of each condition detected by the sensors, which were established according to the optimal conditions. According to Figure 5.13, it is possible to verify that the fans were on for almost the entire period during cultivation to maintain the temperature as established since the ambient temperature during these periods was lower. On the other hand, the heaters were barely activated, as shown in Figure 5.14.

Manjericão

AUTOMATIC MODE

Temperature 20

HumidityAir 75

HumiditySoil 75

Luminosity 100

Manter em condições normais

O cultivo se encontra em condições normais.

Faltam 5 dias para o fim do cultivo.

LEDs: 10d 00:38:46

Coolers: 17d 07:56:21

Atomizer: 0d 20:30:50

Valves: 0d 01:05:14

Heater: 0d 02:23:40

Figure 5.12: Monitoring of parameters in real time by Node-RED.

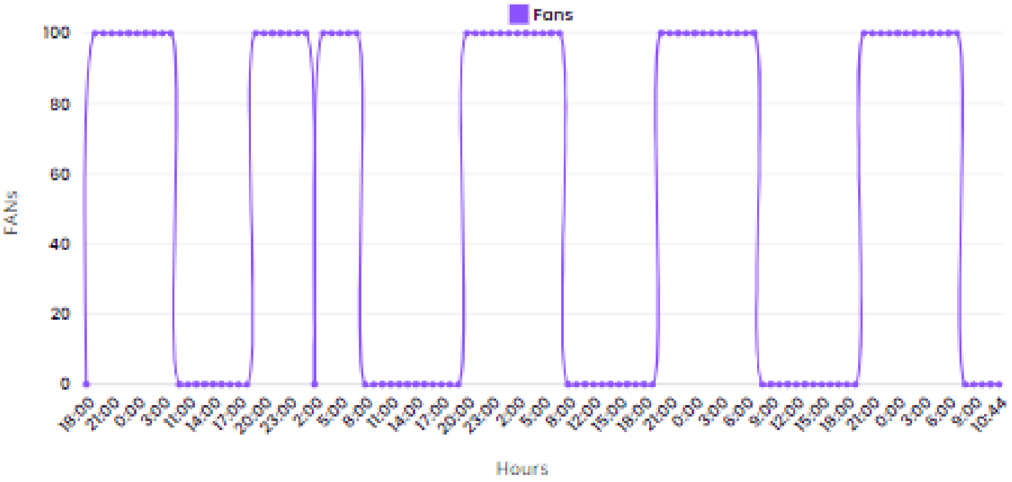


Figure 5.13: Fan performance over time during cultivation.

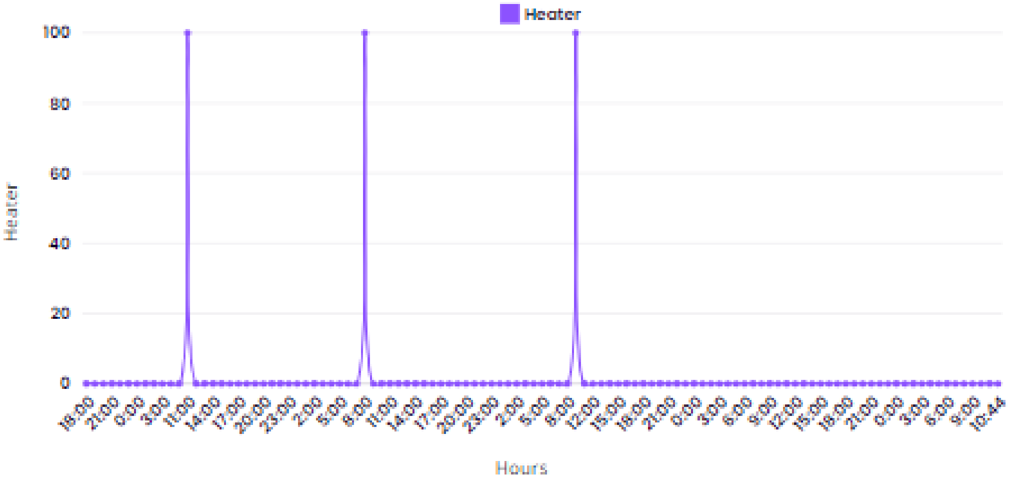


Figure 5.14: Heater performance over time during cultivation.

The irrigation systems were standardized according to the soil moisture values, obtained by the sensors, and the atomizers, in turn, were activated only when the ventilated ones were deactivated. For this reason, the atomizers were activated in short periods, as shown in Figure 5.16.

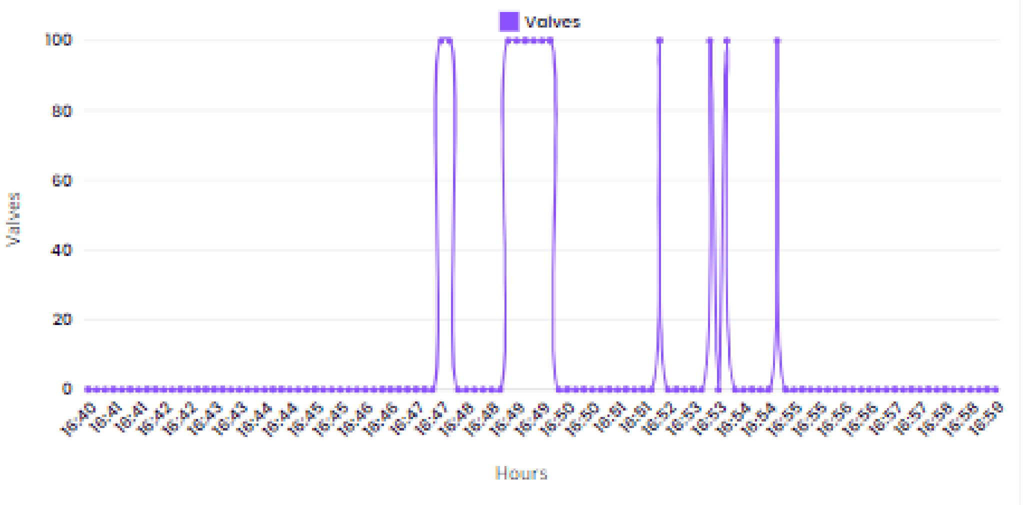


Figure 5.15: Performance of micro sprinklers over time during cultivation.

Figure 5.12 shows the monitoring of parameters made remotely by accessing a Raspberry Pi or web pages connected to the IP address. On the other hand, the systems allow local access to OLED displays for direct monitoring of the local module, as shown

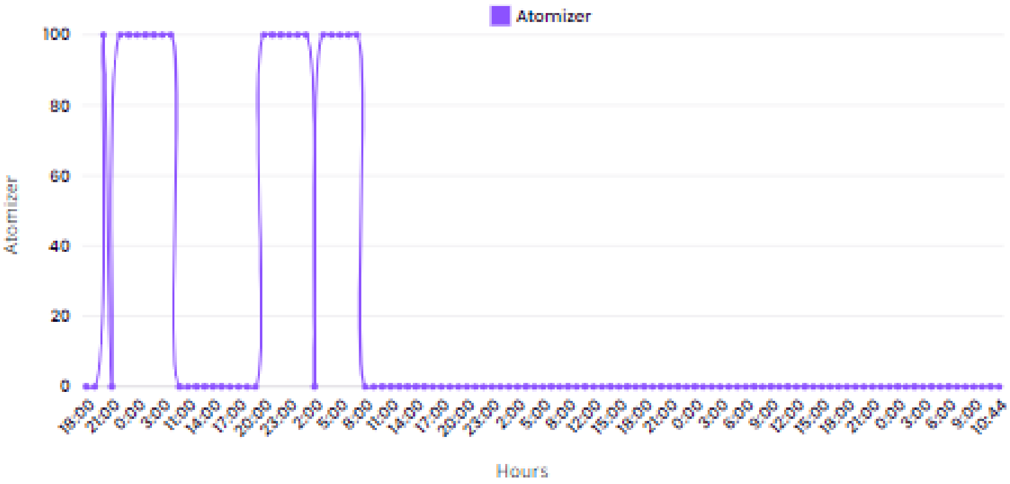
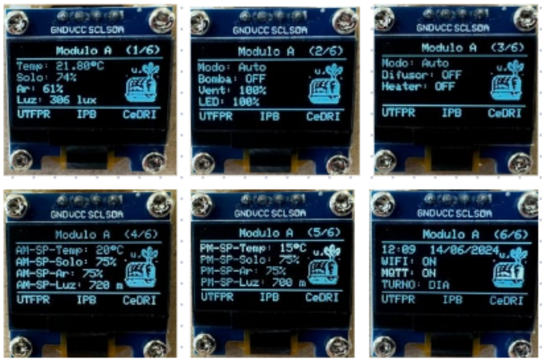


Figure 5.16: Atomizer performance over time during cultivation.

in Figures 5.17 and 5.18, respectively.

Figure 5.17: Remote monitoring by Raspberry Pi.



```
Modulo A  (1/6)
Temp: 21.80°C
Solo: 74%
Ar: 61%
Luz: 306 lux
UTFPR   IPB   CeDRI
```

```
Modulo A  (2/6)
Modo: Auto
Bomba: OFF
Vent: 100%
LED: 100%
UTFPR   IPB   CeDRI
```

```
Modulo A  (3/6)
Modo: Auto
Difusor: OFF
Heater: OFF
UTFPR   IPB   CeDRI
```

```
Modulo A  (4/6)
AM-SP-Temp: 20°C
AM-SP-Solo: 75%
AM-SP-Ar: 75%
AM-SP-Luz: 720 m
UTFPR   IPB   CeDRI
```

```
Modulo A  (5/6)
PM-SP-Temp: 15°C
PM-SP-Solo: 75%
PM-SP-Ar: 75%
PM-SP-Luz: 700 m
UTFPR   IPB   CeDRI
```

```
Modulo A  (6/6)
12:09  14/06/2024
WIFI: ON
MQTT: ON
TURNO: DIA
UTFPR   IPB   CeDRI
```

Figure 5.18: Local monitoring located on the module via OLED display.

Thermal accumulation was accounted for from the beginning of cultivation since the complete basil period was established at 28 days for harvest, with 7 days of germination. The cycle was maintained under normal conditions for standard monitoring in the first culture tested.

The Figures 5.19, 5.20 and 5.21 illustrate different stages of microgreen growth. Finally, the harvest was carried out within the established period. According to Figure 5.21, the crops present the illustrated characteristics with a length of, on average, 2 cm, which does not correspond to the expected size of 3 cm.

Figure 5.19: First stages of basil growth.

Figure 5.20: Intermediate stages of basil.

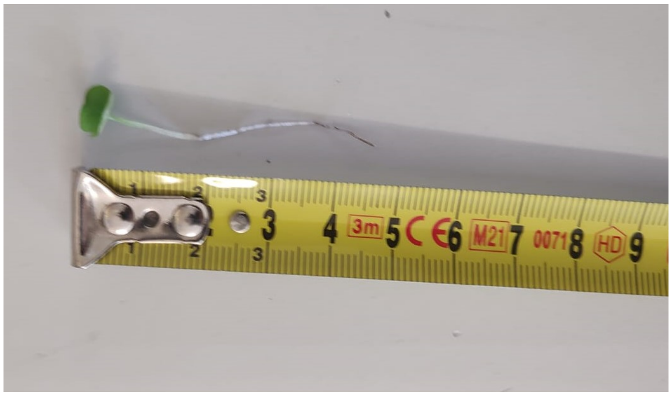


Figure 5.21: Harvested basil microgreen.

Chapter 6

Conclusion and Future Work

Environmental conditions have a significant impact on conventional agriculture, directly affecting the production cycle of microgreens. Despite attempts to mitigate these effects, some metrics can still be optimized according to the parameterization of problems associated with environmental instabilities. Furthermore, cultivation cycles vary depending on the availability of resources and harvest forecasts depend on the stability of these factors.

As a response to these challenges, this project aims to remotely monitor resource management and the behavior of crops in indoor environments, seeking a sustainable alternative in this scenario. These practices, with the introduction of monitoring technologies, in addition to optimizing the use of resources, promote more efficient, flexible and continuous production.

In this context, monitoring and forecasting technologies with MQTT protocols and JSON data format, associated with IoT technologies, provided structured communication between systems, capable of receiving and sending information in real time. According to this information, the algorithms established the optimal conditions for basil cultivation, monitored the development of cultivation under these conditions and made harvest forecast estimates.

Based on the results obtained, it was possible to develop the proposed system and the monitoring of internal environments has been shown to be satisfactory in relation to specific conditions. The operating times of the actuators were in accordance with the needs of the crops, which demonstrates the climatic adaptations made according to ideal conditions. Furthermore, the estimates presented a more accurate prediction about the end of cultivation as the microgreens developed within the stipulated period. The results also demonstrated that monitoring technologies in this scenario promoted the reduction of the effects caused by climatic conditions. In this way, operating costs were reduced as crop efficiency was improved.

In this sense, local and remote monitoring was essential to monitor the parameterization and automation of the greenhouses. The environmental conditions in the indoor environment were as expected, showing few fluctuations during the analyzed period. Furthermore, the forecasting algorithms established accurate estimates as the harvest was carried out on the scheduled date.

Finally, to improve indoor environment monitoring systems, some improvements considered from the development of this work can be integrated into future work. Improvements considered include the implementation of additional sensors to monitor other considerable parameters such as Carbon Dioxide (CO_2) sensors and water level sensors, the migration of the remote interface to cellular applications, the integration of voice commands for more dynamic interactions, the introduction of new independent modules to expand cropping systems, among others.

In this work, monitoring factors such as temperature, humidity, and light were fundamental for the development of microgreens. However, there are other factors that also influence physiological activities, such as the level of CO_2. As CO_2 is used during photosynthesis, including sensors to monitor and control this parameter would be a differentiator.

For the continued production of microgreens, it is essential to ensure that the water reservoir is never empty as this would consequently interrupt irrigation, harming plant growth. For this reason, the integration of water level sensors into the system is promising, ensuring that irrigation systems operate without interruptions.

To improve accuracy in forecasting statistics, it is necessary to consider all factors that directly affect the productive development of microgreens. In this context, measuring all parameters that influence maturation, in addition to thermal accumulation calculations, would be essential.

To facilitate remote management of parameters, migrating the graphical interface to cellular applications would allow for more exclusive and standardized monitoring. In this context, the interface could also be integrated with voice commands, streamlining user interaction during system monitoring.

Furthermore, it would be interesting to adopt a continuous cycle approach so that databases are generated exclusively for each crop. In this way, the system can only contain information during a period to be analyzed under a given production. When the algorithm identifies the end of a cycle, the system must archive the database and generate another file or page for the next one.

Finally, these systems can also be complemented with new independent modules to increase crop projection across different crops. To achieve this, it would be necessary to replicate the protocols and implementation of components that allow the system to continue across a set of elements.

Bibliography

[1] G. Ebissa, K. Yeshitela, H. Desta, and A. Fetene, “Urban agriculture and environmental sustainability,” *Springer Nature*, Mar. 2023.

[2] J. Allred and N. Mattson, *Growing better greenhouse microgreens in under control: tips for controlled environment growing*, Vegetable Growers News. Bari: Greenhouse Product News, 2018, pp. 10–13.

[3] U. S. D. of Agriculture, “Cultivating better greenhouse microgreens under controlled environment: Tips for controlled environment cultivation,” Tech. Rep., 2022.

[4] M. S. Farooq, S. Riaz, M. Abu Helou, F. S. Khan, A. Abid, and A. Alvi, “Internet of things in greenhouse agriculture: A survey on enabling technologies, applications, and protocols,” *IEEE Access*, Apr. 2022.

[5] C. Maraveas and T. Bartzanas, “Application of internet of things (iot) for optimized greenhouse environments,” *AgriEngineering*, vol. 3, p. 4, 2021. DOI: `https://doi.org/10.3390/agriengineering3040060`.

[6] J. K. Lone, R. Pandey, and Gayacharan, “Microgreens on the rise: Expanding our horizons from farm to fork,” *Heliyon*, vol. 10, p. 15, 2024. DOI: `https://doi.org/10.1016/j.heliyon.2024.e25870`.

[7] F. D. Gioia and P. Santamaria, *Microgreens, agrobiodiversity and food security*, (eds). Microgreens. Bari: Eco-logicaeditore, 2015, pp. 7–24.

[8] J. S. Lee, W. G. Pill, B. B. Cobb, and M. Olszewski, "Seed treatments to advance greenhouse establishment of beet and chard microgreens," *The Journal of Horticultural Science and Biotechnology*, vol. 79, p. 570, 2004. DOI: `https://doi.org/10.1080/14620316.2004.11511806`.

[9] J. J. V. Müller and V. J. Vizzotto, "Manejo do solo para a produção de hortaliças em ambiente protegido," *Revista Brasileira de Agrometeorologia, Santa Maria*, vol. 20, pp. 32–35, Dec. 1999.

[10] D. Treadwell, R. Hochmuth, L. Landrum, and W. Laughlin, "Microgreens: A new specialitycrop," *University of Florida, IFASExtension*, Jun. 2021.

[11] P. R. G. Pereira and H. E. P. Martinez, "Produção de mudas para o cultivo de hortaliças em solo e hidroponia," *Informe Agropecuário*, vol. 20, pp. 24–31, Dec. 1999.

[12] D. Almeida, *Manual de Culturas Hortículas Vol.1*, 3° Edição. Portugal: Editorial Presença, 2006, p. 360.

[13] D. de Faria Albernaz Vieir, *CATÁLOGO BRASILEIRO DE HORTALIÇAS*, Plano Mídia Comunicação. Brasília: Embrapa Hortaliças, 2010, pp. 10–13.

[14] L. M. Aguiar and J. M. Guiscem, "Graus-dia estimado com diferentes valores de temperatura base na cultura do milho (zea mays i.)," *XXIV Congresso Nacional de Milho e Sorgo*, Sep. 2022.

[15] A. Prela and A. M. de Arruda Ribeiro, "Determinação de graus-dia acumulados e sua aplicação no planejamento do cultivo de feijão-vagem (phaseolus vulgaris l.) para londrina-pr," *Revista Brasileira de Agrometeorologia, Santa Maria*, vol. 10, pp. 83–86, May 2002.

[16] N. de Safras, "Fenologia – estádios fenológicos," Tech. Rep., 2021.

[17] M. Azure, "O que é um servidor cloud?" Tech. Rep.

[18] S. Madakam, R. Ramaswamy, and S. Tripathi, “Internet of things (iot): A literature review,” *Journal of Computer and Communications*, vol. 3, pp. 164–173, 2015. DOI: `http://dx.doi.org/10.4236/jcc.2015.35021`.

[19] P. Gokhale, O. Bhat, and S. Bhat, “Introduction to iot,” *International Journal of Advanced Research in Science, Engineering and Technology*, vol. 5, pp. 41–44, 2018. DOI: `10.17148/IARJSET.2018.517`.

[20] S. Kumar, Kanchan, A. Kumar, P. Agarwal, and H. Maurya, “Internet of things (iot) applications and challenges: A review,” *International Journal of Engineering Sciences Emerging Technologies*, vol. 11, pp. 359–367, 2023.

[21] M. Azure, “Microsoft, tecnologias e protocolos de iot,” Tech. Rep., 2023.

[22] F. C. D. S. SILVA, “Modelos cliente-servidor do modbus tcp e publish-subscribe do mqtt: Quando utilizar cada um deles e quais suas vantagens e desvantagens?” *NOVUS*, Jan. 2019.

[23] O. Foundation, “Node-red: Programação low-code para aplicações orientadas a eventos,” Tech. Rep., 2023.

[24] S. Ian, *Engenharia de Software*, 9st. Rua Nelson Francisco, 26, BRASIL: Pearson Education, 2011, ISBN: 978-85-7936-108-1.

[25] A. B. Júnior and S. L. Honório, “Local de instalação e construção de estufas para cultivo de hortaliças,” *Informe Agropecuário*, vol. 20, pp. 11–14, Dec. 1999.

[26] S. R. Martins, H. S. Fernandes, F. N. de Assis, and M. E. G. Mendez, “Caracterização climática e manejo de ambientes protegidos: A experiência brasileira,” *Informe Agropecuário*, vol. 20, pp. 15–23, Dec. 1999.

[27] K. Benke and B. Tomkins, “Future food-production systems: Vertical farming and controlled-environment agriculture,” *Sustainability: Science, Practice and Policy*, vol. 13, pp. 13–26, 2017. DOI: `https://doi.org/10.1080/15487733.2017.1394054`.