SMART BOTANICA: AI-DRIVEN MULTI-SPECIES PLANT CARE ASSISTANCE AND MANAGEMENT SYSTEM

Mrs. Trupti Gangakhedkar [1], Dharshan Kumar C [2], Nithya Shridhara Bhat [3], Srajan [4], Yashavantha H S [5]

[1] *Assistant Professor, Dept of ISE, East West Institute of Technology, Bengaluru*
[2,3,4,5] *Student, Dept of ISE, East West Institute of Technology, Bengaluru*

Abstract - Smart Botanica represents an innovative system that harnesses artificial intelligence and the Internet of Things to transform the management of plant care. By merging cutting-edge technologies with intuitive user interfaces, this project utilizes a blend of IoT sensors, AI algorithms, and real-time data analytics to track essential factors such as soil moisture, nutrient content, and overall plant health. The system features AI-driven image recognition capabilities to identify plant diseases and pest issues, offering precise diagnoses along with actionable treatment suggestions. Among the notable features of Smart Botanica are automated irrigation, disease forecasting, resource optimization, and comprehensive health reporting. Its modular architecture allows for scalability, making it suitable for both small home gardens and extensive agricultural operations. The user-friendly interface promotes interaction, enabling users to monitor plant conditions, obtain care recommendations, and create performance reports. Furthermore, the system includes an AI-powered query-handling module that enhances user engagement and facilitates knowledge exchange. By integrating automation, accuracy, and sustainability, Smart Botanica effectively tackles contemporary challenges in plant care, including resource inefficiency, slow disease identification, and the knowledge gap among users. This document outlines the system's architecture, methodology, and testing outcomes, illustrating its capacity to revolutionize plant management practices and foster sustainable agricultural methods.

Key Words*: *Python, Tkinter-GUI, Arduino MEGA 2560, Open CV, Scikit-learn, PySerail, MySQL, DHT 22, NPK Sensor, IR Sensor, Moisture Sensor.

1.INTRODUCTION

The cultivation of medicinal and edible plants has long been fundamental to sustainable living; however, it currently encounters considerable obstacles in our rapidly evolving society. Numerous individuals find it challenging to effectively manage plant care due to constraints on time, insufficient knowledge, and the intricate requirements of various plant species. Conventional methods frequently prove inadequate, resulting in neglect, improper maintenance, and the forfeiture of the distinctive health and nutritional benefits these plants provide. Furthermore, problems such as overwatering, nutrient mismanagement, and delayed identification of diseases or pests lead to suboptimal growth and waste of resources. The practice of cultivating medicinal and edible plants has long been fundamental to sustainable living. This comprehensive approach ensures efficient, sustainable plant management.

Numerous individuals find it challenging to effectively care for their plants due to constraints on time, insufficient knowledge, and the intricate requirements associated with various plant species. Conventional methods frequently prove inadequate, resulting in neglect, improper maintenance, and the forfeiture of the distinctive health and nutritional benefits these plants provide. Furthermore, problems such as overwatering, nutrient mismanagement, and delayed identification of diseases or pests lead to subpar plant growth, increased expenses, and waste of resources.

In addition, current plant care solutions tend to offer disjointed support, concentrating on particular elements like irrigation or disease detection while lacking a holistic, integrated strategy. This disconnection forces users to depend on multiple tools, complicating the plant care process and making it less accessible for novices or those managing a variety of plant types. The absence of data-driven insights and practical guidance further intensifies these difficulties, hindering the achievement of consistent and optimal plant health.

To tackle these urgent challenges, there is an increasing need for an intelligent, automated, and user-friendly system that not only streamlines plant care but also guarantees accuracy, sustainability, and adaptability. The initiative, Smart Botanica, introduces an innovative solution aimed at revolutionizing plant care management through the incorporation of artificial intelligence (AI), the Internet of Things (IoT), and data-driven insights.

Smart Botanica employs IoT-enabled sensors to continuously monitor essential parameters such as soil moisture, temperature, and nutrient levels. The data gathered is analyzed by AI models trained on extensive datasets, facilitating accurate diagnosis of plant diseases, early identification of pests, and customized nutrient management recommendations. The system features a user-friendly interface.

To tackle these issues, there is a pressing need for an intelligent, automated system that streamlines plant care while ensuring accuracy and sustainability. This initiative, Smart Botanica, proposes an AI-driven, IoT-enabled solution aimed at transforming plant management. By incorporating cutting-edge technologies, it presents a user-friendly platform that automates critical tasks such as irrigation, disease detection, and nutrient management.

The system employs IoT sensors to continuously monitor parameters such as soil moisture, temperature, and nutrient levels. These data points are processed by AI models trained on extensive datasets, facilitating the early detection of diseases and pests. By providing actionable recommendations and tailored care plans, the system empowers users to make informed decisions, irrespective of their previous experience in plant care.

Smart Botanica distinguishes itself through its comprehensive approach, merging automation, precision, and sustainability. It is designed to accommodate a diverse array of users, from amateur gardeners to professional agriculturists, offering a scalable and adaptable solution. This paper delves into the design, implementation, and potential impact of Smart Botanica, highlighting its capacity to revolutionize plant care practices and foster eco-friendly agriculture.

1.1 LITERATURE SURVEY

Recent developments in plant care technology have resulted in groundbreaking solutions for more effective and efficient gardening practices. Doe and Smith [1] concentrated on the application of image recognition technology within plant care tools to accurately identify plant species and monitor their growth. Their findings highlight the incorporation of computer vision into gardening, facilitating meticulous plant observation for both amateur and professional gardeners. In a similar vein, Tanaka, Wong, and Lee [2] investigated the use of IoT sensors for wireless monitoring of plant health, allowing for real-time assessment of soil moisture, temperature, and light conditions, thereby improving agricultural productivity and sustainability.

Significant advancements have also been made in IoT-driven automation. Gupta and Sharma [3] created an indoor gardening system that automates the management of watering and lighting to optimize plant growth, utilizing smart technology to cater to the varying needs of different plants. Gardena [4] launched a commercial smart gardening system that combines sensors, irrigation, and weather data for holistic garden management. This system exemplifies the integration of IoT into consumer gardening products, enhancing user convenience.

The role of artificial intelligence in agriculture has also seen substantial growth. Yang and Nguyen [5] illustrated how machine learning algorithms can assess plant health by evaluating environmental and visual data, providing targeted solutions for disease management. Furthermore, Garden.org [6] offers a community-oriented platform that leverages a database and shared knowledge to empower gardeners, demonstrating how online resources can enhance the impact of technological innovations in gardening.

Kumar and Patel [7] examined comprehensive support systems for the care of multiple plant species, highlighting the difficulties and potential solutions associated with managing various types of plants within a unified framework. Concurrently, Rajan [8] pointed out the drawbacks of parameter-based diagnostics in assessing plant health, advocating for the development of sophisticated algorithms to address the limitations of conventional approaches.

Li and Wang [9] focused on the challenge of ambiguous disease diagnoses in plants, promoting the use of specific indicators to enhance identification and management processes. In a related initiative, Taylor [10] underscored the significance of improving user query resolution in automated gardening systems, with the goal of increasing their user-friendliness and effectiveness.

Singh and White [11] considered the human aspect of gardening technologies, emphasizing the necessity for specialized training for individuals operating AI-driven systems. Rao and Bansal [12] investigated the obstacles to providing flexible solutions for multi-species care.

Green and Carter [13] stressed the importance of developing diverse care protocols to accommodate a broader array of plant species, suggesting a framework for adaptable gardening systems. In the area of user engagement, Johnson [14] examined the impact of interactive reporting in gardening applications, aiming to enhance user satisfaction and overall system usability.

The paper by Chatzopoulos, Papoutsidakis, and Sofianos [15] focuses on developing a smart plant pot system that leverages Arduino-based technology and internet connectivity for automated plant care. The system integrates various sensors and actuators to monitor and control critical parameters such as soil moisture, temperature, and light exposure, ensuring optimal plant growth.

2. SYSTEM ARCHITECTURE

The Smart Botanica system features a sophisticated layered and modular architecture that seamlessly integrates IoT-enabled hardware, advanced software, and user-focused interfaces to provide an effective solution for plant care. At the foundation, the hardware layer consists of various IoT sensors, such as soil moisture sensors, NPK sensors, DHT sensors (for temperature and humidity), and a camera module, which gather real-time data regarding soil and environmental conditions. These sensors are linked to an Arduino Mega 2560 microcontroller, which processes the collected data and facilitates communication with the software layer. The software layer incorporates AI models specifically trained for detecting diseases and pests, conducting soil analysis, and offering personalized care recommendations.

These models employ machine learning techniques, including Decision Tree Classifiers and k-Nearest Neighbors (kNN), to ensure precise diagnostics. The user interface, crafted using Python's Tkinter library, allows users to monitor their plants in real-time, gain actionable insights, and engage in interactive query handling. Data management is supported by a MySQL database, which ensures efficient storage and retrieval of plant health records and care histories. This architecture is both modular and scalable, allowing for the integration of additional sensors or features, thus making it suitable for both small home gardens and extensive agricultural operations. By merging hardware, AI-driven analytics, and intuitive interfaces, Smart Botanica provides a seamless, efficient, and sustainable approach to plant care.

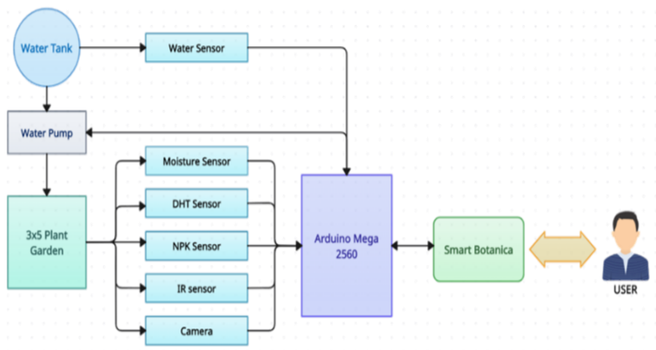

Fig 1: System Architecture of Smart Botanica

The Smart Botanica: AI-Driven Multi-Species Plant Care System utilizes cutting-edge technologies to enhance plant management through the effective integration of IoT sensors, artificial intelligence algorithms, and an engaging user interface.

The system initiates the process by gathering critical environmental data, such as soil moisture levels and nutrient content (NPK), via IoT sensors. These sensors provide continuous monitoring of the plant's condition, relaying real-time information to the central system. The collected data is subsequently analyzed by AI algorithms, which assess the plant's health and identify potential issues, including diseases or pest infestations, through image recognition and environmental cues. This immediate analysis facilitates the prompt identification and diagnosis of any health concerns, allowing for timely intervention. After processing the data, the system produces actionable insights and recommendations based on the findings. In instances where pests or diseases are identified, the system offers customized treatment options tailored to the specific problem.

The outcomes are presented to the user through a detailed report and an interactive interface, which also functions as a platform for ongoing engagement. The report delivers an in-depth overview of the plant's health, growth patterns, and care history, enabling users to monitor progress and make well-informed decisions. Furthermore, the interactive interface empowers users to ask questions or seek guidance regarding plant care, ensuring they receive personalized assistance tailored to their individual requirements.

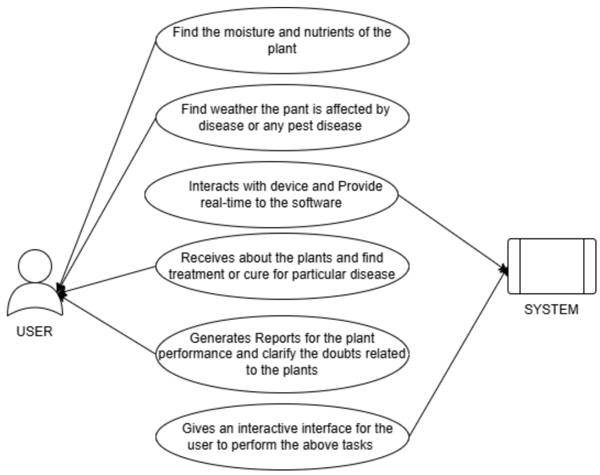

Fig 2: UML Diagram of Smart Botanica

HARDWARE USED

1.Arduino MEGA 2560

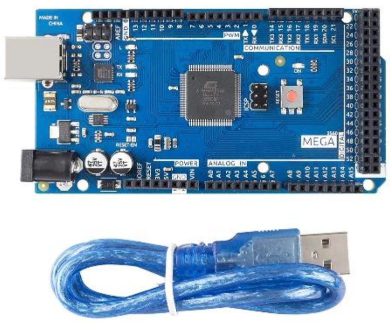

Fig 3: Arduino MEGA 2560

The Arduino Mega 2560 is a microcontroller board that utilizes the ATmega2560 chip. It features 54 digital input/output pins, 16 analog inputs, 4 hardware UARTs, and 256 KB of flash memory. With 8 KB of SRAM and a clock speed of 16 MHz, it is capable of handling complex applications such as robotics, extensive displays, and sensor systems. Its enhanced input/output capabilities and memory capacity render it suitable for large-scale projects and tasks that require multiple peripherals. Fully compatible with the Arduino IDE, it facilitates programming and integration with a diverse array of libraries and shields, making it an excellent choice for both enthusiasts and professionals.

2. Soil Moisture Sensor

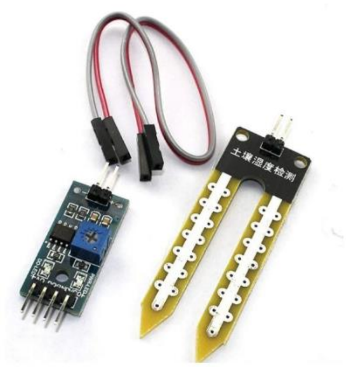

Fig 4: Soil Moisture Sensor

A Soil Moisture Sensor is designed to assess the water content within the soil, which is crucial for various applications in agriculture, gardening, and environmental monitoring. These sensors generally operate using either resistive or capacitive probes. Resistive sensors gauge the resistance between two probes, where a lower resistance signifies a higher moisture level, whereas capacitive sensors measure variations in capacitance caused by the presence of water. The data produced by the sensor can be either analog or digital, facilitating automated irrigation systems, monitoring of soil health, and efficient water management. These sensors are compatible with microcontrollers such as Arduino and Raspberry Pi, providing capabilities for real-time monitoring and data logging. Some models even incorporate built-in temperature sensors to offer further insights into soil conditions.

3. DHT Sensor

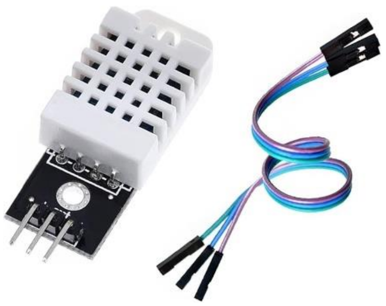

Fig 5: DHT Sensor

The DHT sensor is a digital instrument designed to measure both temperature and humidity, frequently utilized in weather monitoring stations, HVAC systems, and applications.

It comprises a thermistor and a capacitive humidity sensor, ensuring precise measurements. The DHT11 is a more economical variant, capable of measuring temperatures between 0 and 50°C and humidity levels ranging from 20 to 80%. In contrast, the DHT22 provides enhanced accuracy, with a temperature range of -40 to 80°C and humidity levels from 0 to 100%. Both sensors can be easily integrated with microcontrollers such as Arduino and Raspberry Pi, facilitating real-time monitoring and data acquisition.

4. NPK Sensor

Fig 6: NPK Sensor

An NPK sensor is designed to assess the levels of Nitrogen (N), Phosphorus (P), and Potassium (K) present in soil or water, which are vital for the growth of plants. By evaluating electrical conductivity, this sensor provides an estimate of nutrient availability, thereby supporting precision agriculture and facilitating informed decisions regarding fertilization. Frequently utilized in smart farming applications, NPK sensors connect with microcontrollers such as Arduino or Raspberry Pi to enable real-time monitoring. The operation of the sensor is generally based on electrochemical or conductivity principles, allowing for the accurate detection of nutrient concentrations, which is crucial for maintaining optimal crop health.

5. IR Sensor

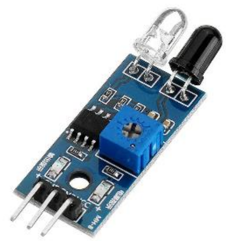

Fig 7: IR Sensor

An infrared (IR) sensor is designed to detect infrared radiation, which remains imperceptible to the human eye but can be identified by specialized detectors. This technology quantifies the intensity of infrared light and is utilized in various applications, including motion detection, temperature assessment, and proximity sensing. They are categorized into two types: active IR sensors, which emit infrared light and measure its reflection, and passive IR sensors, which identify radiation emitted by objects, frequently employed in motion detection scenarios.

6. Camera Module

Fig 8: Camera Module

The Lapcam LWC-042 is a small, high-definition camera module tailored for embedded systems and Internet of Things (IoT) applications. It is equipped with a wide-angle lens that allows for the capture of both video and still images, rendering it suitable for applications such as object detection, surveillance, and machine vision. This module is compatible with microcontrollers including Arduino and Raspberry Pi, facilitating straightforward integration and serial communication for real-time visual input in robotics, smart home technologies, and other uses. Its compact design and user-friendly interface make it a flexible solution for a range of projects.

SOFTWARE USED

1. Python

Python is a high-level, interpreted programming language recognized for its ease of use and clarity. Introduced in 1991 by Guido van Rossum, it employs indentation to delineate code blocks, which improves readability and lowers the barrier to entry for new learners. The language boasts a comprehensive standard library that facilitates various tasks, including file management, networking, and regular expressions. Additionally, Python's extensibility enables smooth integration with other programming languages. The extensive selection of third-party libraries accessible through the Python Package Index (PyPI) enhances its adaptability for diverse applications, such as web development, data analysis, artificial intelligence, and scientific computing.

2. Tkinter

Tkinter serves as the standard library in Python for developing graphical user interfaces (GUIs), allowing developers to create desktop applications that incorporate interactive components such as windows, buttons, and menus. Built upon the Tk GUI toolkit, it is user-friendly and comes pre-installed with Python, eliminating the need for any extra configuration. Tkinter provides a variety of customizable widgets, including buttons, labels, and text boxes, and operates on an event-driven model, which makes it particularly suitable for effortlessly creating interactive, cross-platform desktop applications.

3. Scikit-Learn

Scikit-learn is a freely available machine learning library for Python, developed on the foundations of NumPy, SciPy, and matplotlib. It offers a range of effective tools for data mining, analysis, and machine learning.

Including algorithms for classification, regression, clustering, and dimensionality reduction. Renowned for its straightforward and uniform application programming interface (API), Scikit-learn structures its features around fundamental concepts such as estimators, transformers, and predictors, facilitating user experimentation with algorithms and enhancing the efficiency of the model development, training, and evaluation processes.

4. OpenCV

OpenCV, or Open Source Computer Vision Library, is a prominent open-source library utilized for real-time computer vision and image processing. Initially created by Intel, it facilitates various tasks including object detection, image manipulation, video analysis, and facial recognition. Developed in C++, OpenCV offers bindings for languages such as Python and Java, ensuring cross-platform accessibility. The library comprises a range of pre-built functions for operations like filtering, edge detection, and feature extraction, as well as advanced methodologies including object tracking and machine learning-based image recognition, rendering it ideal for applications in fields such as robotics, healthcare, automotive, and security.

5. PySerail

PySerial is a Python library designed to facilitate communication with hardware devices through serial ports, accommodating protocols such as RS-232, RS-485, and TTL. This library is compatible across multiple platforms, including Windows, macOS, and Linux. PySerial allows for straightforward configuration and data transfer with minimal coding, enabling functionalities such as adjusting baud rates, managing flow control, and setting timeouts. Its user-friendly nature and adaptability render it suitable for both novice and experienced developers engaged in serial communication for embedded systems and sensor applications.

6. MySQL

MySQL is a widely recognized open-source relational database management system (RDBMS) that employs SQL for the management of data. Renowned for its speed, dependability, and user-friendliness, MySQL is extensively utilized in web development, particularly within the LAMP stack. It offers essential features such as data integrity, ACID compliance, and scalability, rendering it appropriate for both small-scale applications and large enterprise systems. MySQL operates across various platforms and includes tools like MySQL Workbench for graphical management, as well as command-line utilities for more experienced users. Its strong support for multiple users, along with advanced indexing and security features, establishes it as a trustworthy option for managing structured data in high-performance environments.

IMPLEMENTATION AND RESULTS

The methodology employed for the intelligent plant care system adopts a systematic framework that integrates sensor technology, machine learning, and Internet of Things (IoT) solutions to facilitate the automation of plant care activities. The initial phase focused on requirement analysis, during which the primary challenges associated with the care of multiple plant species were identified.

These challenges encompassed disease identification, nutrient management, and the automation of irrigation processes. Prioritizing these features was essential to ensure the system's capability to effectively monitor plant health, regulate soil nutrients, and automate care tasks.

In terms of hardware configuration, a variety of sensors were incorporated into the system to assess both plant and environmental conditions. Monitoring included soil moisture levels, NPK (Nitrogen, Phosphorus, Potassium) content, and environmental parameters such as temperature and humidity, measured using a DHT sensor. Additionally, a camera was integrated to capture images of the plants, aiding in disease detection. The Arduino Mega 2560 was selected as the microcontroller due to its capacity to connect with multiple sensors and control various actuators, including watering pumps and fans.

Data collection involved the compilation of datasets related to plant diseases, pest images, and soil nutrient levels. The dataset concerning plant diseases was utilized for training machine learning algorithms, while real-time data from the sensors facilitated the monitoring of soil and environmental conditions. This aggregated data served as the foundation for AI-driven decision-making and automation within the system.

For the development of the AI model, two machine learning algorithms were utilized: a Decision Tree Classifier and k-Nearest Neighbors (kNN). The Decision Tree was specifically trained to identify plant diseases and pests through images captured by the camera, while the kNN model was employed to predict diseases based on the similarity of images. Both models underwent optimization for accuracy through various data pre-processing techniques, including resizing and augmentation.

The frontend development focused on creating a user-friendly graphical user interface (GUI) using Tkinter in Python. This interface presents real-time data regarding plants, such as soil moisture, temperature, and nutrient levels, and provides care recommendations derived from the collected information. Furthermore, the system generates historical reports to monitor plant health over time.

In the backend development, Python was utilized for data processing, executing AI model predictions, and managing automation tasks driven by sensors. The backend interacts with the Arduino Mega to regulate watering systems according to soil moisture levels, while also initiating environmental adjustments based on temperature and humidity readings.

To facilitate IoT integration, Wi-Fi modules, such as the ESP8266, were implemented to connect the system to the cloud, enabling remote monitoring and control. A Node-RED dashboard was established to visualize sensor data and allow users to engage with the system from a distance. This integration guarantees that the system remains accessible from any location and provides real-time updates.

Testing and simulation were conducted in two distinct phases. Initially, hardware components underwent evaluation through Proteus and Tinker CAD simulations to verify accurate sensor readings and actuator responses. Subsequently, software testing was executed via unit tests and integration tests to confirm the operational integrity of the AI models, data processing, and user interface.

For deployment, the backend and database of the system were hosted on cloud platforms such as AWS or Microsoft Azure.

This approach facilitated scalability, remote accessibility, and dependable data backup, thereby enhancing the system's robustness and its ability to manage extensive datasets and multiple users.

Lastly, maintenance is a continuous endeavor that encompasses regular updates to the AI models in response to new data, sensor calibration, and feature improvements. Consistent system updates and the incorporation of user feedback ensure that the plant care system remains both effective and efficient over time.

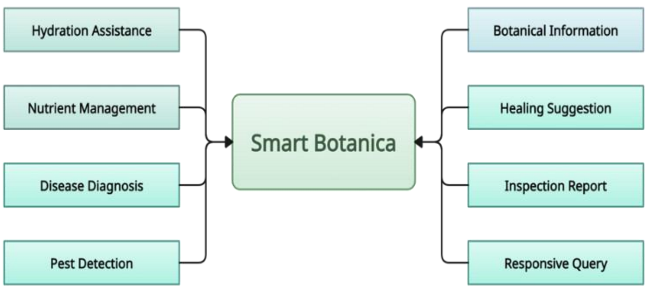

Fig 9: Software Design of Smart Botanica

The architectural framework of the plant care system is meticulously organized to facilitate smooth integration among data acquisition, AI-based decision-making, user engagement, and automation functionalities. This framework encompasses several tiers, such as the data processing tier, AI algorithms, user interface, and backend management. The main objective is to develop a resilient and intuitive system capable of automating plant care activities, continuously monitoring plant health, and delivering precise feedback to the user.

RESULTS

Fig 10: Overall Model Representation

A Plant Care Assistance System is an innovative solution aimed at aiding users in the proper maintenance of their plants, encompassing indoor houseplants, outdoor garden varieties, and specific crop types. This system is capable of incorporating diverse technologies to assess plant health, deliver care guidelines, and suggest strategies for achieving optimal growth.

The Plant Care Assistance System generally comprises integrated software and hardware elements that collaborate to oversee, work together to monitor, maintain, and provide guidance for plant care. Also sustain, and offer recommendations for the care of plants.

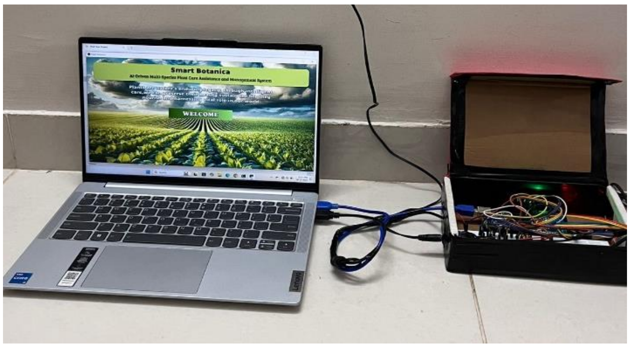

Fig 11: Software and Hardware Setup

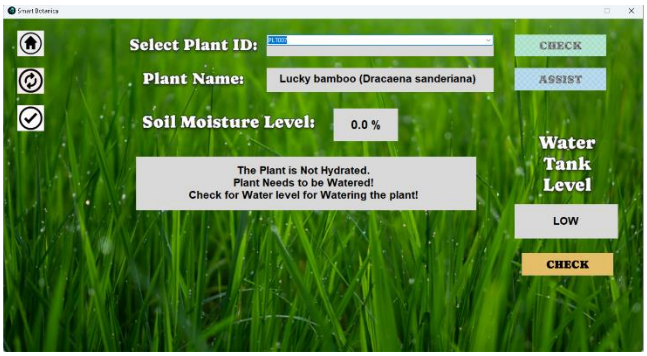

Fig 12: Output of Hydration Assistance

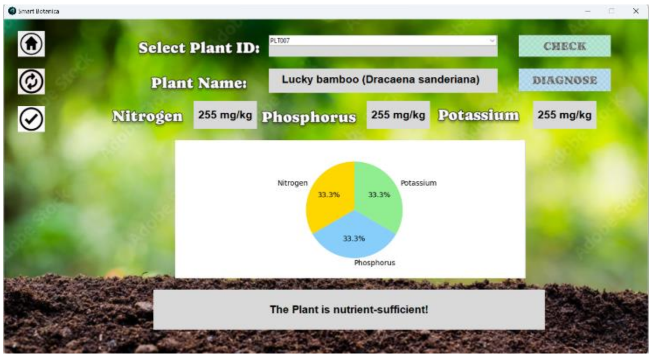

Fig 13: Output of Nutrients Management

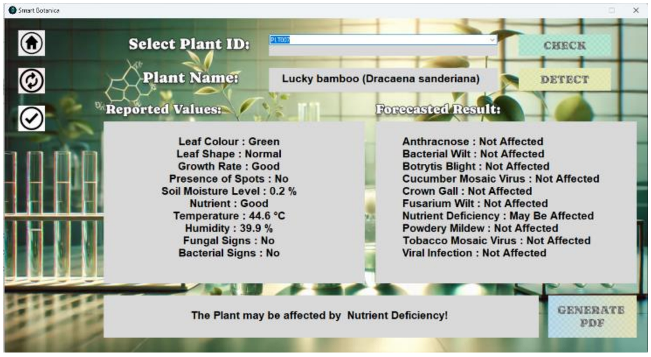

Fig 14: Output of Disease Diagnosis

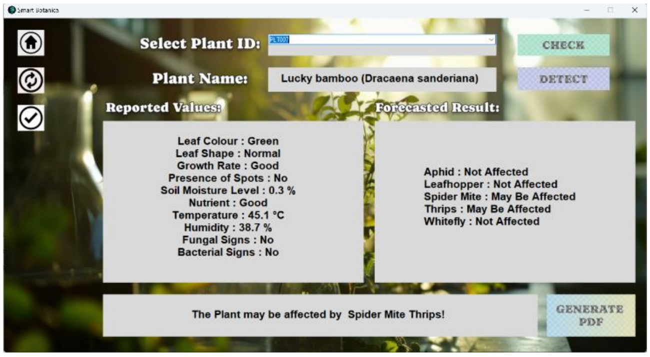

Fig 15: Output of Pest Detection

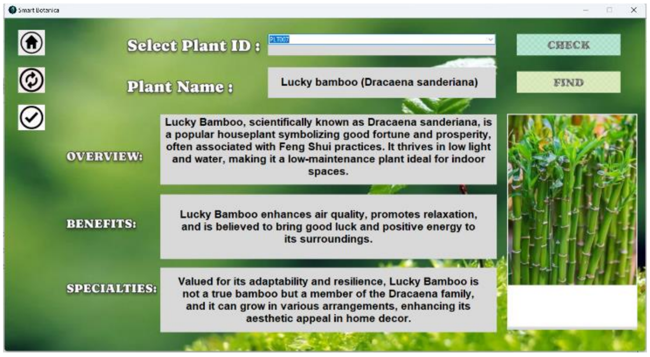

Fig 16: Output of Botanical Information

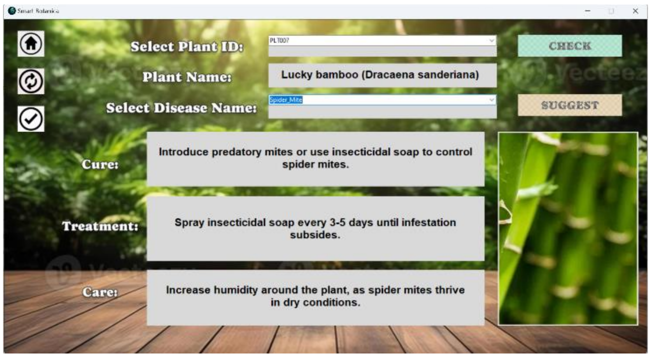

Fig 17: Output of Healing Suggestion

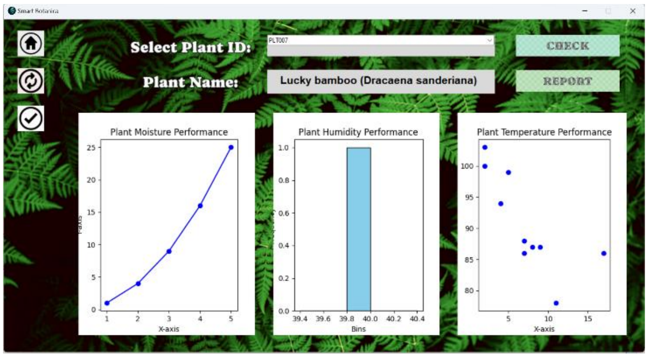

Fig 18: Output of Inspection Report

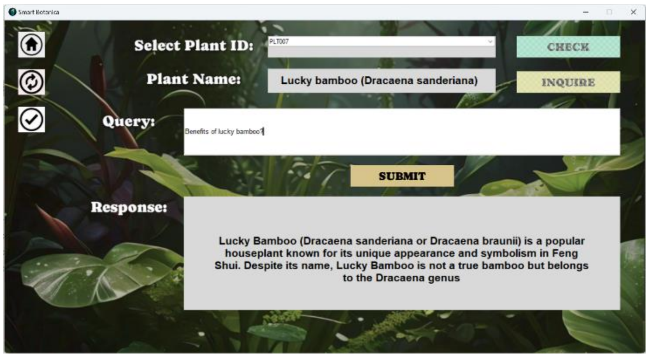

Fig 19: Output of Responsive Query

The "Smart Botanica" system enhances the management of plant care through various features, including hydration support, nutrient oversight, disease identification, and pest monitoring. By utilizing real-time data and artificial intelligence, it offers tailored recommendations for watering, fertilization, and pest management. Users can explore detailed plant profiles, obtain recovery suggestions, and monitor plant health through performance reports. Additionally, the system provides prompt responses to inquiries, delivering customized guidance on plant care. This all-encompassing strategy promotes effective and sustainable plant management. These features create a holistic, user-friendly platform for effective and sustainable plant care.

3. CONCLUSIONS

The "Smart Botanica" initiative utilizes artificial intelligence and the Internet of Things to streamline the care of plants, encompassing tasks such as irrigation, disease identification, pest monitoring, and nutrient regulation. Its intuitive system fosters both sustainability and scalability, rendering plant care more efficient and widely accessible. By providing real-time monitoring, AI-generated insights, and tailored recommendations, it improves decision-making processes and supports environmental health. This project exemplifies the potential of technology to revolutionize plant care and advance sustainable practices.

REFERENCES

[1] J. Doe and A. Smith, "Plant Care Applications: Image Recognition for Gardening," International Journal of Plant Technologies, vol. 5, no. 2, pp. 34-42, 2020.

[2] M. Tanaka, K. Wong, and S. Lee, "Wireless monitoring of plant health using IoT sensors," IEEE Transactions on Smart Agriculture, vol. 12, no. 4, pp. 23-30, Dec. 2021.

[3] P. Gupta and R. Sharma, "Automated indoor gardening systems: Enhancing growth conditions with IoT," in Proceedings of the International Conference on Smart Systems and Agriculture, Jaipur, India, pp. 98-102, 2019.

[4] Gardena, "The Gardena Smart System for Integrated Gardening," Gardena Official Site, 2023. [Online]. Available: https://www.gardena.com. [Accessed: Dec. 6, 2024].
[5] L. Yang and H. Nguyen, "Machine learning-based plant health diagnostics," Journal of Artificial Intelligence in Agriculture, vol. 8, no. 1, pp. 45-60, 2022.
[6] Garden.org, "Community-driven gardening resources and plant database," Garden.org Official Site, 2023. [Online]. Available: https://www.garden.org. [Accessed: Dec. 6, 2024].
[7] A. Kumar and N. Patel, "Towards holistic support systems for multi-species plant care," IEEE Smart Agriculture Systems, vol. 6, no. 3, pp. 70-85, 2021.
[8] J. S. Rajan, "Challenges in parameter-based diagnostics for plant health," Environmental Technology Journal, vol. 10, no. 7, pp. 120-134, 2020.
[9] H. Li and P. Wang, "Improving disease diagnosis in plants through precise indicators," Transactions of Plant Data Science, vol. 4, no. 2, pp. 12-19, 2019.
[10] S. B. Taylor, "Addressing user queries in automated gardening systems," Smart Gardening Research Papers, vol. 3, no. 4, pp. 56-65, 2020.
[11] R. K. Singh and J. White, "Training requirements for AI-driven gardening systems," Journal of Human-Technology Interaction, vol. 5, no. 2, pp. 42-48, 2021.
[12] V. Rao and T. Bansal, "Multi-species plant care systems: Challenges and solutions," IEEE Agriculture and Environmental Systems Journal, vol. 7, no. 5, pp. 89-100, 2021.
[13] E. M. Green and D. Carter, "Diversifying protocols for multi-species plant care," Agricultural Informatics Journal, vol. 8, no. 1, pp. 15-30, 2022.
[14] P. Johnson, "Improving engagement through interactive reporting in gardening apps," Human Factors in Digital Agriculture, vol. 9, no. 3, pp. 105-118, 2023.
[15] A. Chatzopoulos, M. Papoutsidakis, and G. Sofianos, "An Automated Plant Pot Controlled via the Internet based on Arduino Applications," International Journal of Agricultural Automation and IoT Systems, vol. 4, no. 1, pp. 12-21, 2024.