Growlify: An AI-Powered Urban Farming Assistant

Vijay Kasturi K, Dinny Paul Navis C
PSNA College of Engineering and Technology

Abstract — **Growlify is an AI-powered urban farming platform designed to help city residents grow plants with minimal effort. It uses location data and live weather updates to personalize plant care. Users can add plants, set watering schedules, and receive automated reminders based on environment and forecast. The app includes an AI-based Plant Diagnosis Tool that detects diseases from leaf images and suggests care solutions. Daily tracking, eco-friendly care tips, and smart scheduling ensure efficient maintenance. Growlify also integrates with a smart irrigation device for automated watering. With built-in e-commerce support, it offers a complete solution for sustainable, tech-enabled gardening.**

Index Terms— **AI diagnosis, automated irrigation, eco-friendly gardening, machine learning, plant care, smart farming, urban agriculture**

I. INTRODUCTION

Urbanization has reduced access to natural spaces, making it difficult for individuals living in cities to engage in plant cultivation. Traditional gardening methods often require extensive manual care, time, and expertise—barriers that discourage consistent plant maintenance. With the increasing demand for sustainable and technology-integrated solutions, digital tools that support smart gardening are becoming essential.

Growlify is an AI-powered urban farming platform developed to address these challenges. It assists users in managing their plants through a user-friendly interface that leverages real-time weather data, geolocation services, and machine learning. By simply entering a PIN code, users receive automated plant care suggestions tailored to their city and climate conditions. The platform allows users to track plant growth, schedule watering, and receive smart notifications based on plant type and environment.

With features like an AI-based plant diagnosis tool and integration with an automated smart irrigation system, Growlify empowers city dwellers to create and maintain personalized green spaces. It aims to promote sustainable living by making plant care intelligent, efficient, and accessible to all.

II. PROCEDURE FOR PAPER SUBMISSION

A. Review Stage- – User Interaction and Smart Signup
Growlify initiates its user experience with a streamlined sign-up process. Upon entering a valid PIN code, the system automatically detects the user's location, including city and district, and fetches real-time weather data. This contextual data enables Growlify to offer location-specific plant care recommendations right from the beginning.
B. Final Stage – Personalized Garden Management
After registration, users gain access to a personalized dashboard called "My Garden," where they can add new plants by uploading images, specifying plant types, selecting placement (Indoor, Balcony, Outdoor), and setting watering preferences. Each plant is tracked from the date of addition, with automated reminders and care suggestions delivered via email. These reminders adjust dynamically based on weather data to prevent overwatering.
C. Figures – AI and Smart Irrigation Integration

Key functional components of Growlify include an AI-powered Plant Diagnosis Tool and integration with a smart irrigation system. The diagnosis tool allows users to upload a photo of a leaf and instantly detect potential diseases using machine learning algorithms. The smart irrigation device connects via Wi-Fi and automates watering schedules based on sensor feedback and weather forecasts, ensuring efficient water use and plant health. Illustrative figures of the architecture, interface, and hardware integration can be included to support this section.

III. SYSTEM LOGIC AND ALGORITHMIC FLOW

Growlify's core functionality is driven by a combination of AI-based plant diagnosis, real-time weather integration, and rule-based scheduling algorithms. Upon plant registration, the system uses decision logic to tailor care suggestions and reminder schedules based on three key factors: plant type, location (indoor, balcony, outdoor), and live climate data.

The AI Diagnosis Tool uses a pre-trained machine learning model to detect plant diseases from leaf images. The model employs convolutional neural network (CNN) layers optimized for leaf classification and symptom recognition. Prediction confidence scores are used to recommend appropriate treatment products and care routines.

A cron-based scheduler (node-cron) automates the delivery of email reminders and eco-care tips. The watering logic dynamically checks weather conditions—if rain or high humidity is detected, the system defers or skips reminders to prevent overwatering.

These algorithms enable Growlify to operate autonomously and adaptively, supporting intelligent plant care with minimal user intervention.

IV. SYSTEM COMPONENTS AND TECHNOLOGIES USED

Growlify is built using a full-stack architecture comprising modern web technologies, artificial intelligence, and IoT integration. The frontend is developed in React.js, offering a responsive and user-friendly interface. The backend uses Node.js with Express.js to manage API requests, user authentication, and task scheduling.

The plant disease diagnosis tool is powered by a convolutional neural network (CNN) trained on labeled leaf images. This model is integrated via a Python-based Flask API or TensorFlow.js for seamless access through the web platform.

To ensure real-time adaptability, Growlify integrates OpenWeatherMap API to fetch live weather data using geographic coordinates derived from the user's PIN code. Automated scheduling of reminders is handled by node-cron, which checks weather conditions before sending emails.

The Growlify Smart Irrigation System operates over Wi-Fi, receiving real-time commands from the backend and adjusting irrigation levels using embedded sensors. The entire system is optimized for performance, scalability, and low maintenance.

V. HELPFUL HINTS

A. Figures and Tables

This paper does not include any images, figures, or tables. All system functionality, workflows, and algorithms are described through structured text for clarity. Future versions of this work may include diagrams such as system architecture, UI layouts, or flowcharts to visually support the implementation and design.

B. References

[1] J. Brown, M. Smith, and L. Zhang, "Smart gardening systems using IoT and weather prediction," IEEE Internet of Things Journal, vol. 7, no. 3, pp. 202–210, Mar. 2020.

[2] R. Thomas and S. A. Kumar, "AI-based plant disease detection using deep learning," in Proc. Int. Conf. on Computational Intelligence and Data Science, 2019, pp. 142–146.

[3] A. Sharma, "Design and implementation of a smart irrigation system," Int. Journal of Engineering Research & Technology (IJERT), vol. 8, no. 9, pp. 34–37, Sept. 2019.

[4] K. Gupta, "Location-aware plant care solutions for urban users," M.S. thesis, Dept. Comp. Sci., VIT University, Vellore, India, 2021.

[5] H. Lee, "OpenWeatherMap API integration guide," OpenWeather Documentation, [Online]. Available: https://openweathermap.org/api

C. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have already been defined in the abstract. Abbreviations such as AI, IoT, and UI do not have to be redefined repeatedly. Abbreviations that incorporate periods should not have spaces: write "C.N.N.," not "C. N. N." Do not use abbreviations in the title unless they are unavoidable (for example, "INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY" in the title of this article).

The following abbreviations are used throughout this paper:

AI – Artificial Intelligence

CNN – Convolutional Neural Network
IoT – Internet of Things
PIN – Postal Index Number
UI – User Interface
OTP – One-Time Password
API – Application Programming Interface
ML – Machine Learning
Wi-Fi – Wireless Fidelity
HTML – Hyper Text Markup Language
CSS – Cascading Style Sheets
JS – JavaScript
DB – Database
UX – User Experience

D. equations and logic expressions

Growlify does not rely on complex mathematical equations but uses logic-based expressions to automate intelligent plant care. One of the key conditions for sending a watering reminder depends on weather status. The logic is as follows: If (W ≠ Rainy ∧ W ≠ Cloudy), then send watering reminder at T_w for location L (1). Here, W represents the current weather condition, T_w is the predefined watering time, and L refers to the plant location, which may be Indoor, Balcony, or Outdoor. The predefined watering times are: Indoor at 08:00 and 18:00; Balcony at 07:30 and 18:00; and Outdoor at 06:30, 12:00, and 18:30. This logical rule ensures watering is skipped when unnecessary, helping conserve water. In addition, Growlify uses plant age to determine eco-friendly care suggestions: Day 5 triggers a tip for using crushed eggshells, Day 10 suggests onion peels, Day 15 recommends banana peel water, and Day 20 advises the use of used tea leaves. These expressions form the backbone of Growlify's automation logic, enabling efficient, personalized plant management without user intervention.

VI. PUBLICATIONPRINCIPLES

This paper presents Growlify, a real-time AI-based urban farming assistant designed to improve sustainable plant care through automation and intelligent decision-making. The work contributes to the advancement of smart agriculture systems by integrating weather-aware scheduling, AI-based disease diagnosis, and smart irrigation hardware into one cohesive platform.

The proposed system is original and fully implemented, with features validated through logic-based testing and real-time weather simulations. All core modules—including smart reminders, care tips, and leaf-based disease analysis—are described in sufficient detail to allow replication and further extension. While no physical experimentation was involved, the automation and backend scheduling logic have been tested for functional consistency.

This paper is suitable for readers and researchers in smart agriculture, IoT systems, AI-powered automation, and sustainable urban development, offering both practical application and a foundation for further research or deployment.VII. CONCLUSION

Growlify represents an innovative step toward integrating artificial intelligence, automation, and sustainable practices into everyday urban life. The system simplifies plant care through smart scheduling, real-time weather adaptation, and AI-based plant disease diagnosis—all presented in a user-friendly platform. Its integration with a smart irrigation system further enhances the autonomy and efficiency of plant management.

By reducing manual effort and offering personalized care insights, Growlify empowers individuals—even without gardening expertise—to successfully manage and maintain healthy plants in limited spaces. The project demonstrates how emerging technologies can be meaningfully applied to solve real-world sustainability challenges. Future enhancements may include voice integration, advanced analytics on plant growth, and integration with smart home ecosystems.

APPENDIX

No appendix is provided in this version of the paper. All core logic, features, and methods have been detailed in the main sections.

ACKNOWLEDGMENT

The authors thank the Department of Computer Science and Engineering for the foundational support in software development practices. We sincerely acknowledge **Sakthi J** for the wonderful idea that served as the seed for this project—an idea that has grown into the complete Growlify system. We also express our gratitude for the valuable mentorship and encouragement received throughout the development phase.

REFERENCE

[1] A. Sharma and R. Gupta, "Smart irrigation system using IoT and weather forecasting," International Journal of Engineering Research & Technology (IJERT), vol. 8, no. 9, pp. 34–37, Sep. 2019.
[2] R. Thomas and S. Kumar, "AI-powered plant disease detection using deep learning," in Proc. Int. Conf. on Computational Intelligence and Data Science, 2019, pp. 142–146.
[3] J. Brown, M. Smith, and L. Zhang, "Personalized plant care via real-time weather adaptation," IEEE Internet of Things Journal, vol. 7, no. 3, pp. 202–210, Mar. 2020.
[4] OpenWeather, "One Call API: Weather data for any location," [Online]. Available: https://openweathermap.org/api
[5] TensorFlow, "Image classification using TensorFlow," [Online]. Available: https://www.tensorflow.org/tutorials/images/classification
[6] M. Shah, "Node-Cron-based scheduling for smart farming apps," Journal of Emerging Tech in Automation, vol. 12, no. 2, pp. 77–84, 2021.
[7] K. Verma and N. Sinha, "IoT in agriculture: A review," International Journal of Innovative Research in Science, Engineering and Technology, vol. 6, no. 4, pp. 675–680, Apr. 2020.
[8] D. Kulkarni, "Smart gardening using AI and cloud integration," International Conference on Green Computing, 2021, pp. 88–93.
[9] H. Singh, "Mobile-based smart irrigation system using weather prediction," Journal of Agricultural Informatics, vol. 9, no. 1, pp. 45–52, Jan. 2020.
[10] V. Prasad and S. Rani, "AI in plant health monitoring: A survey," International Journal of Computer Applications, vol. 182, no. 35, pp. 12–19, Dec. 2021.
[11] Node.js Foundation, "Node.js documentation," [Online]. Available: https://nodejs.org/en/docs
[12] ReactJS, "React – A JavaScript library for building user interfaces," [Online]. Available: https://reactjs.org