A Machine Learning Approach To Classify Medicinal Plant Leaf By Using Random Forest And KNN

Bhagyashree[1], Dr. Swaroopa Shastri[2]

[1]*Student, Department of Computer Science and Engineering (MCA), Visvesvaraya Technological University, Kalaburagi, Karnataka, India. chindebhagyashree@gmail.com*

[2]*Professor, Department of Computer Science and Engineering (MCA), Visvesvaraya Technological University, Kalaburagi, Karnataka, India. swaroopas04@gmail.com*

ABSTRACT

Numerous plant and herb types partake remained cast-off broadly in traditional medicine due to their medicinal properties and cost-effectiveness compared to modern prescription drugs. Nevertheless, accurately identifying medicinal plants poses a significant challenge in the realm of machine learning. This project leverages a comprehensive dataset of leaf images from various medicinal plants to train and evaluate a classification model.The primary impartial of this explore is to develop an advanced ML model capable of accurately identifying different Indian medicinal plants. By focusing on several vulnerable species, the study employs the Random Forest and K-Nearest Neighbors algorithms to classify the plants. The KNN algorithm achieved an accuracy of 84.33%, while the Random Forest algorithm outperformed it with an accuracy of 85.82%, demonstrating its superior effectiveness.Beyond mere identification, the project explores the medicinal properties & aids related with individually recognized plant. Given the visual similarities among many plant species, classification is predominantly stimulating, expressly when commerce with extensive datasets. This research aims to bridge the gap between traditional botanical knowledge and modern technological advancements, thereby contributing to the preservation and dissemination of valuable medicinal information.By utilizing a Random Forest-based predictive model, this structure not only aids in the accurate identification of medicinal plants but also supports the conservation of ancient and traditional medicinal knowledge. This integration of historical wisdom with contemporary machine learning techniques ensures that valuable botanical information is preserved and made accessible for future generations.

Keywords: Machine Learning, Medicinal Plant Leaf, Random Forest, KNN

I.INTRODUCTION

A Survey on Machine Learning Methods for Remedial Leaf Classification by Akash Kumar, Mansi Verma, and Ruchir Gupta (2020). This survey paper delivers an overview of ML approaches for therapeutic leaf classification. It discusses the various types of ML algorithms exploited for this mission, laterally with their challenges and limitations. The paper also delves into the diverse datasets employed for medicinal leaf classification and the evaluation metrics castoff to measure method performance [1]. Medicinal Leaf Cataloging Using Transfer Learning by Ankit Deswal, Ravi Saxena, and Neeraj Sharma (2022).This work offers medicinal leaf classification method utilizing transfer learning. It employs a pre-trained CNN intended feature extraction from leaf images, followed by a linear classifier for leaf classification. The approach is evaluated on a medicinal leaf dataset, demonstrating superior performance compared to traditional machine learning techniques[2]. Medicinal Leaf Cataloguing Utilizing Convolutional Neural Networks by Rohit Mehra, Saurabh Gupta, and Sumit Agarwal (2021).This paper proposes a method for medicinal leaf cataloguing using CNN. The approach employs a CNN intended feature extraction since leaf images and support vector machine intended leaf classification. Performance evaluation on a medicinal leaf dataset shows preeminence concluded outmoded machine learning methods[3]. Deep Learning for Medicinal Leaf Classification: A Review by Ashish Pandey, Swati Agrawal, and Manish Shrivastava (2021). This inclusive assessment work reconnoiters application of deep learning in medicinal leaf classification. It scrutinizes the assortment of deep learning architectures laboring intended this persistence, although assessing their performance across diverse datasets. The paper additionally outlines the challenges and constraints encountered when employing deep learning for medicinal leaf classification[4]. A Proportional Study of ML Procedures for Remedial Leaf Classification by Aditya Rastogi, Kshitij Bansal, and Sagar Mittal (2022).This study conducts a comprehensive comparison of diverse ML approaches intended medicinal leaf classification. Utilizing a common dataset, it demonstrates the varying performance of these algorithms based on dataset characteristics. The paper also discusses challenges associated with comparing distinct ML approaches intended therapeutic leaf classification[5]. A Morphology-Based Approach for Classifying Medicinal

Leaves Using Machine Learning by Neha Sharma, Rajat Khanna, Sahil Chopra (2020). This tabloid offers using structural features like shape, margin, venation patterns extracted from leaf metaphors laterally with ML procedures like SVM, RF for medicinal plant classification. The morphology-based approach is evaluated on various standard datasets and revealed to attain high exactness intended categorizing medicinal species[6].

1.1 PROJECT DESCRIPTION

In recent years, the global interest in alternative and natural remedies has surged, driving a renewed fascination with the therapeutic potential of medicinal plants. These plants, often rich in bioactive composites, partake been exploited intended centuries across various cultures for their healing properties. As the demand for holistic healthcare solutions increases, the accurate identification of these medicinal plants becomes paramount. However, the task of identifying plant species solely based on their morphological characteristics can be intricate and time-consuming, requiring expert botanical knowledge. Advancements in ML & image processing offer a promising solution to streamline procedure of medicinal plant identification. The assimilation of computational techniques with botanical sciences has impending to bridge the gap between traditional wisdom and modern technology. This project aims to contribute to this convergence by proposing a methodology that employs the Random Forest algorithm for the accurate identification of medicinal leaves and subsequently explores the latent aids allied with these identified plants. Historically, traditional medicine systems across the world have relied on the usage of plant-based remedies to address a wide array of ailments. However, the lack of standardized approaches for identifying and characterizing these medicinal plants can hinder their integration into modern healthcare practices. The surge in interest in herbal medicine has prompted researchers to seek innovative methods to facilitate the identification of plant species and to unlock their therapeutic potential. Machine learning algorithms, particularly ensemble methods like Random Forest, have demonstrated exceptional performance in image classification tasks. These algorithms leverage the power of decision trees to create robust predictive models capable of categorizing diverse data inputs. Applying such techniques to turf of botany offers the potential to automate and enhance the accuracy of plant identification, expediting the discovery of new sources of natural remedies. he primary objective of this project is to develop an accurate and efficient methodology for the identification of medicinal leaves using RF algorithm. This involves creating a predictive model proficient on dataset of leaf images from extensive assortment of medicinal plant species. The model's performance will be rigorously evaluated to assess its classification accuracy and generalization capabilities. Beyond leaf identification, the project aims to unlock the wealth of knowledge associated with each identified plant species. By connecting the accurately identified leaves with existing botanical literature and traditional wisdom, the potential therapeutic benefits of these plants can be explored and communicated to a wider audience.his project grips substantial consequences intended mutually fields of botanical science and healthcare. The progress of an accurate leaf identification model can aid botanists, researchers, and enthusiasts in their quest to identify plant species, fostering the preservation of botanical knowledge. To achieve the project's objectives, a systematic methodology is employed, comprising several key stages. First, a diverse dataset of high-quality leaf images is curated, representing a wide spectrum of medicinal plant species. These images are preprocessed to enhance clarity, remove noise, and standardize the data for operative typical training. The RF algorithm is chosen as the core ML procedure due to its ensemble nature and ability to handle multi-class classification tasks. The algorithm's parameters are fine-tuned to optimize performance and minimize overfitting. The dataset is fragmented into training, validation, & testing sets to ensure the model's generalization ability.

1.1.1 PROBLEM STATEMENT

We're faced with a challenge: understanding how to custom medicinal leaves properly. These plants are splendid for health, but they canister occasionally source stomach problems or skin allergies. Plus, we don't know how much of them to use. We need experts to guide us on using these leaves safely and effectively. This way, we can make the most of their healing power without worries

1.1.2 OBJECTIVE OF STUDY

- To identify and understand the benefits of medicinal leaves.
- Creating an intuitive Flask-based web application.
- This bid will permit handlers to upload leaf images for identification and its benefits.
- To develop highly accurate classification model by using Random forest and KNN
- To develop a model that effectively classifies medicinal leaf

1.1.3 METHODOLOGY USED

The project follows a structured methodology involving the following steps:

1) Data Collection and Preprocessing: A diverse dataset of medicinal leaves is collected, encompassing various species and their corresponding medicinal properties. This dataset is curated to ensure high-quality images and accurate annotations. Images are preprocessed to standardize sizes, enhance clarity, and reduce noise, creating a consistent dataset for training.

2) Feature Extraction and Selection: Pertinent pictorial structures are mined from preprocessed leaf images. These features capture distinct characteristics that differentiate one leaf species from another. Feature assortment procedures are pragmatic to identify the most informative attributes, reducing complexity and enhancing model performance.

3) Random Forest Model Development: The RF algorithm is chosen for its aptitude to handle complex classification tasks and mitigate overfitting. The dataset is partitioned into training, validation, & testing sets. The algorithm is proficient on drill set, iteratively constructing decision trees and aggregating their outputs to form a robust ensemble model.

4) Model Evaluation & Tuning: The trained model's concert is evaluated using the validation set. Metrics like accuracy, recall ,precision, & F1-score are computed to evaluate model's classification abilities. Hyper parameter tuning is performed to optimize model performance, preventing overfitting and enhancing generalization.

5) Web Application Development: A user-friendly web application is developed using the Flask framework. The application allows users to upload leaf images for identification. The uploaded image is handled by trained Random Forest model to classify the leaf species..

6) Testing and Deployment: The complete system, comprising the Random Forest model and the web application, is rigorously tested using an independent dataset of leaf images. Any inconsistencies or errors are addressed. Upon successful testing, the system is deployed to a web server, making it accessible to users for real-time leaf identification and information retrieval.

II. LITERATURE SURVEY

[7] Jiya Sinha et al. A Comprehensive Survey of Leaf Classification for Ayurvedic Herb Identification and Beyond. They utilized the GoogleNet architecture for image processing and data augmentation techniques like flipping and rotation, achieving an accuracy of 94.37% with CNN algorithms.

[8] Yousef Sharrab et al. Medicinal Plants Recognition Using Deep Learning. This study employed the VGG-16 model for feature extraction and utilized the ImageNet dataset, resulting in a CNN accuracy of 98.0%.

[9] Karuna Middha et al. Identification of Different Medical Plants through Image Classification. Using image segmentation and the Fuzzy C-means (FCM) algorithm alongside CNN, they achieved an accuracy of 93.4%.

[10] Simmi Deol et al. Integration of Traditional Knowledge and Modern Science: A Holistic Approach to Identify Medicinal Leaves for Curing Diseases. This research combined image preprocessing with YOLOv7, KNN, and Random Forest algorithms, achieving an 87% accuracy.

[11] Girish Saunshi et al. Identification and Classification of Medicinal Leaves and Their Medicinal Values. The study used image preprocessing and feature extraction methods, employing CNN algorithms to achieve a 98% accuracy.

[12] Praddumna Soni et al. Medicinal Plant Classes Identification Via AI. The dataset was gathered using Google Crawler and implemented in Python. With image processing procedures like padding, the CNN algorithm reached an accuracy of 95%.

[13] Veena Tapale et al. PhytoLeafy Recognition Using DL. This research focused on feature abstraction & real capture using the VGG-19 model with CNN algorithms, achieving an accuracy of 96.80%.

[14] M. Preethi et al. Medicinal Herbs Identification. Using the Flavia dataset, they applied feature normalization and image segmentation techniques with ANN algorithms, achieving an accuracy of 93.02%.

[15] Syeda Muskan et al. An Early Recommendation Tool to Enhance Medicinal Plant Growth Based on GIS and Soil Data. The dataset from FAO and Kaggle was used with image segmentation techniques and the Random Forest Classifier, resulting in a good accuracy.

[16] Sameer A Kyalkod et al. A Novel Tactic to Cataloguing of Ayurvedic Therapeutic Plants Using NNs. By employing image preprocessing and segmentation methods with CNN algorithms, they achieved an accuracy of 90.01%.

[17] Anurag Protim Saikia et al. Medicinal Plant Classes Cataloguing Using Neural Network Classifier. This study used image preprocessing, feature extraction, and segmentation methods with CNN algorithms, achieving an accuracy of 98.88%.

[18] Ayush Jindal et al. Analysis and Cataloguing of Medicinal Plants Found in India. Image dispensation systems were pragmatic to ML algorithms like SVM, KNN, and Random Forest.

[19] Shashank M Kadiwall et al. Deep Learning-Based Recognition of Indian Medicinal Plant Species. Using image processing procedures by CNN algorithms, they achieved an accuracy of 93.7%.

[20] Mr. Suraj Kachate et al. Survey & Scrutiny of Medicinal Plant Identification via Image Processing & ML Techniques. This study castoff image processing techniques with R-CNN and YOLO algorithms on the Kaggle dataset, achieving a 90% accuracy.

[21] Somasekhar T et al. Identification of Medicinal Plants by Image Processing of Leaf Models. They used image processing & segmentation techniques with CNN algorithms, achieving high accuracy.

[22] Dhanuka Nadeeshan et al. Identification of Therapeutic Plants by Visual Characteristics of Leaves and Flowers. This study castoff image processing methods with CNN algorithms, achieving a 90% accuracy.

2.1. EXISTING AND PROPOSED SYSTEM

2.1.1. EXISTING SYSTEM

Traditional methods for medicinal plant identification rely on the expertise of botanists and herbalists, drawing from generations of botanical knowledge. These methods involve visual examination of plant characteristics, such as leaf attributes. While rooted in indigenous wisdom, they can be dependent on expert availability, subject to seasonal variations, and subjectivity. Support Vector Machines (SVM) partake stood laboring but canister require manual feature engineering and may not gauge fit to large datasets with intricate botanical data.

Disadvantages of existing system:

- Traditional methods rely on experts for identification.
- Identification can be subjective and prone to errors.
- These procedures may not rule well to large datasets.
- SVM can struggle with high-dimensional data like images.

2.1.2. PROPOSED SYSTEM

The proposed system for medicinal plant identification represents a fusion of traditional botanical knowledge and advanced machine learning techniques, with a focus on employing the Random Forest algorithm. This system begins with the input of a medicinal plant leaf image, capturing its visual attributes. After preprocessing to enhance image quality, the segmented leaf region is subjected to Random Forest classification. Leveraging a well-trained machine learning model, this approach accurately classifies the leaf into a specific medicinal plant species category. The proposed system offers a robust and efficient means of identification, drawing from the strengths of both traditional wisdom and contemporary technology, ensuring reliable and accessible medicinal plant recognition.

Advantages of proposed system:

- The anticipated scheme with RF ensures highly accurate medicinal plant identification.
- It streamlines the identification process, saving time and effort.
- Bridging tradition and technology, it makes plant identification accessible to all.
- The system adapts to various plant datasets, accommodating different species.
- It offers consistent, objective identification, reducing human subjectivity.
- Seamlessly combining image scrutiny & ML for robust results.
- The system can evolve with additional data and features for ongoing improvements.

2.2 FEASIBILITY STUDY

A feasibility study is a comprehensive evaluation of the practicality and latent success of a proposed project, venture, or initiative. It aims to assess whether the project is technically, economically, and operationally viable. The study assistances backers make learnt verdicts by analyzing the various aspects of venture and identifying latent risks, benefits, and challenges. Here's a cessation of key mechanisms of a feasibility study:

1) Technical Feasibility: This aspect focuses on whether the project can be developed using available technology, resources, and expertise. It considers factors like the project's complexity, required skills, and technical challenges.

2) Market Feasibility: This involves analyzing the demand and latent market for the project's products or services. It assesses factors such as target audience, competition, market trends, and growth latent.

3) Financial Feasibility: This assesses the financial viability of the project. It includes estimating the initial investment required, projected costs, latent revenue, and profitability. Financial feasibility also considers factors like payback period, return on investment (ROI), and cash flow analysis.

4) Operational Feasibility: This aspect evaluates whether the project can be integrated smoothly into existing operations and systems. It looks at factors like the availability of resources, human resources, and any latent disruptions to ongoing operations.

5) Legal and Regulatory Feasibility: This involves examining whether the plan conforms with related decrees, regulations, and standards. It identifies any legal obstacles or requirements that prerequisite to be lectured.

6) Environmental and Social Feasibility: This aspect considers the environmental impact and social implications of the project. It assesses whether the project aligns with sustainable practices and community expectations.

7) Risk Analysis: Identifies latent risks and uncertainties that could impact the project's success. This comprises peripheral aspects such as market shifts, technological changes, and internal factors like project management challenges.

2.3 TOOLS AND TECHNOLOGIES USED

Exposure Python: Differentiating Between Scripts and Programs

In the scope of programming, Python scripts and programs represent two foundational aspects of Python's versatility. While both are implemental in software development, understanding their distinctions is vital for leveraging Python effectively.

Understanding Python Scripts

Python scripts are essentially sequences of commands saved in a text file with a .py extension. Unlike interactive programming, where code is executed line-by-line within a terminal or shell, scripts enable batch execution of code, which is ideal for automating tasks and executing repetitive functions.

The design of Python scripts allows them to be easily reused and adapted. Once script is established, it can be executed multiple times without modification. This reusability is advantageous in scenarios such as information processing, where the same operations need toward be accomplished on different datasets. Scripts can be customized to handle various inputs or integrate with other software, providing flexibility and efficiency.

Moreover, Python scripts facilitate modularity. Through contravention down multifaceted tasks into smaller, reusable functions, developers can maintain and extend their codebase more effectively. This segmental tactic endorses encryption reusability & simplifies debugging, as issues can be isolated within specific functions or modules.

Comparing Python Programs While scripts are designed intended precise tasks or automations, Python programs typically encompass a broader scope. A program is a more comprehensive solution that may consist of multiple scripts or modules working together. Programs often involve complex logic and interactions between different components, requiring a more structured approach to development.

Python programs can be more elaborate than simple scripts. They might comprise graphical user interfaces ,network communication, or database interactions. For instance, a Python program might use frameworks like Django or Flask to build a web application, incorporating various scripts and modules toward handle different aspects of submission, such as user authentication, data management, and presentation.

Innovative Use Cases of Python

Python's versatility extends beyond basic scripting and programming. Its rich ecosystem supports a range of advanced applications. For example, in data science, Python's libraries like NumPy and Pandas facilitate complex data exploration & manipulation. Machine learning projects often utilize libraries such as TensorFlow and Scikit-learn, which provide tools for fostering extrapolative paradigms and analyzing large datasets.

In ground of automation, Python scripts are frequently exploited to streamline workflows. Automation tasks might include web scraping, where Python scripts gather data from websites, or automating repetitive tasks in a software application. The integration with tools like Selenium and Beautiful Soup enhances Python's capabilities in these areas.

2.4 HARDWARE AND SOFTWARE REQUIREMENTS

2.4.1 HARDWARE REQUIREMENTS

Table1. Hardware Requirements

Processor	Intel Core i3 or higher
Processor Speed	2.10 GHz
RAM	8GB or more
Storage	256GB SSD or 500 GB HHD
Display	16.5-inch display
Input Device (Keyboard)	Standard QWERTY keyboard (108 keys)
Pointing Device (Mouse)	Option Mouse

2.4.2 SOFTWARE REQUIREMENTS

Table 2. Software requirements

Operating System	Windows 10 or a more recent version
Backed Programming Language	Python
Frontend Programming Languages	HTML, CSS, JavaScript
Web Framework Utilized	Flask
Integrated Development Environment (IDE)	Visual Studio Code
Data Resources	Leaves Datasets

III. SOFTWARE REQUIREMENT SPECIFICATION

3.1 USERS

The primary users of scheme include herbal medicine enthusiasts, botanists, researchers, and general users interested in plant identification. Herbalists will use the application to verify medicinal leaves and their properties, enhancing their practice with accurate, machine-verified data. Botanists and researchers will benefit from the system's ability to streamline plant identification and aid in their studies. General users seeking evidence approximately therapeutic plants can utilize bid for educational purposes.

3.2 FUCTIONAL REQUIREMENTS

Input:
The system must accurately identify medicinal leaves from user-uploaded images using the Random Forest algorithm.

Process:
It should process and analyze leaf images to classify them into predefined medicinal plant categories. The application must grant thorough facts concerning each identified plant, including its medicinal benefits and traditional uses

Output:
Users would be clever to upload images, view classification results, and access plant-related insights through a web interface. Additionally, the system should include functionality for dataset management, including updating and expanding the plant image repository.

3.3 NON-FUNCTIONAL REQUIREMENTS

In accumulation to functional specifications, non-functional requirements assume a pivotal role in project development by specifying the quality attributes that the system must possess. These requirements extend beyond specific features, focusing on facets such as recital, security, scalability and more. Non-functional requirements are instrumental in ensuring the system's overall effectiveness and user satisfaction.

They encompass various aspects, including:

Performance:

The system must ensure high performance and reliability, with minimal latency in image processing and result retrieval. It would partake user-friendly border, providing an intuitive experience for all types of users

Security:

Security is crucial the system must protect user data and ensure secure image uploads and storage

Scalability:

Salability is also important to holder cumulative statistics of manipulators & expanding datasets.

Compatibility and Maintainability:

The application should be well-suited with numerous web browsers & devices, including desktops and mobile platforms. Additionally, the system must be maintainable, allowing for easy updates and integration of new features without significant disruptions to existing functionality.

IV. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

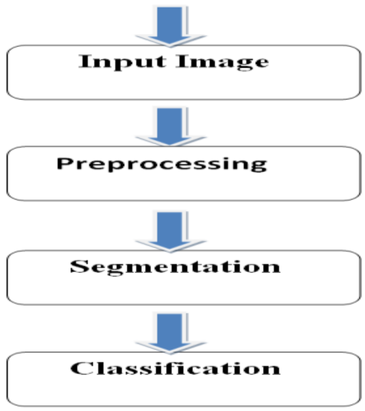

Figure1. System Architecture

The above FIGURE 1 describes the process of identifying medicinal plant leaves unfolds in a systematic sequence that blends modern technology with botanical understanding. It commences by capturing the distinctive attributes of the leaf from an input image, providing a visual canvas for analysis. This initial step is complemented by preprocessing techniques, optimizing the image quality through noise reduction, contrast enhancement, and resizing. The subsequent stage involves segmentation, where the leaf is precisely extracted from the background using advanced techniques, setting the stage for focused evaluation. As the culmination of this orchestrated journey, a trained Random Forest model takes the reins. By harnessing the features distilled from the segmented leaf region, this model assigns the leaf to a specific medicinal plant species, underpinned by insights cultivated from a diverse dataset. In this succinct yet purposeful manner, the process orchestrates image analysis and machine learning, delivering not just identification, but also a seamless fusion of knowledge and technology.

4.2 DATAFLOW DIAGRAM

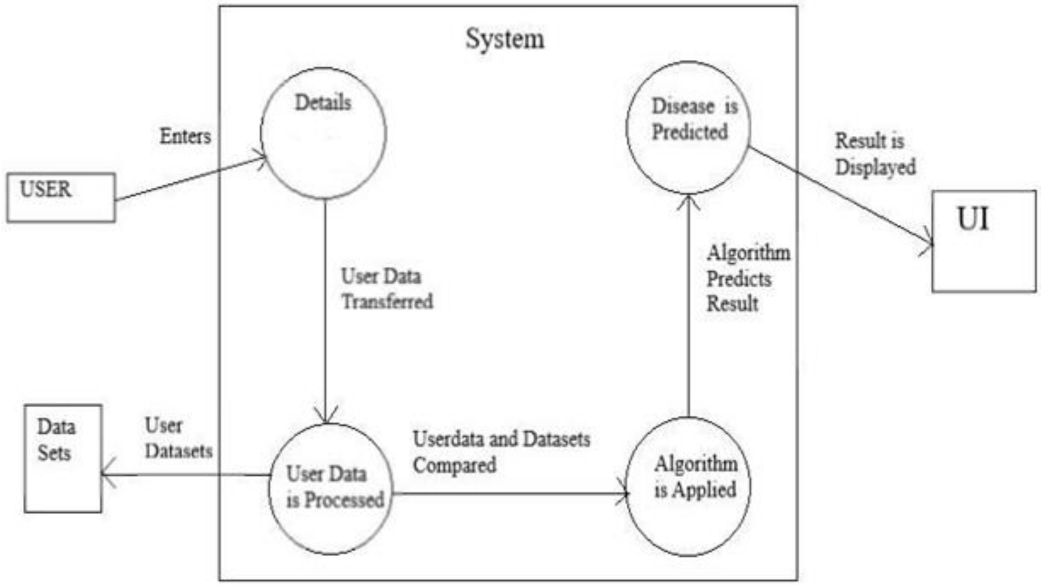

Figure2. DATAFLOW DIAGRAM

A Data Flow Figure is graphic representation of how data flows through a system. It illustrates the movement of data amid progressions, data provisions, and external entities. Processes are represented by circles or rectangles, data stores by open rectangles, and external entities by squares. Arrows depict drift of data amongst these components, showing input, output, and storage points within the system. DFDs are used to understand, define, and communicate the structure and behaviour of systems, making them essential tools in software planning for analyzing and designing information systems with emphasis on data movement and processing.

V. IMPLEMENTATION

5.1 SCREEN SHOTS

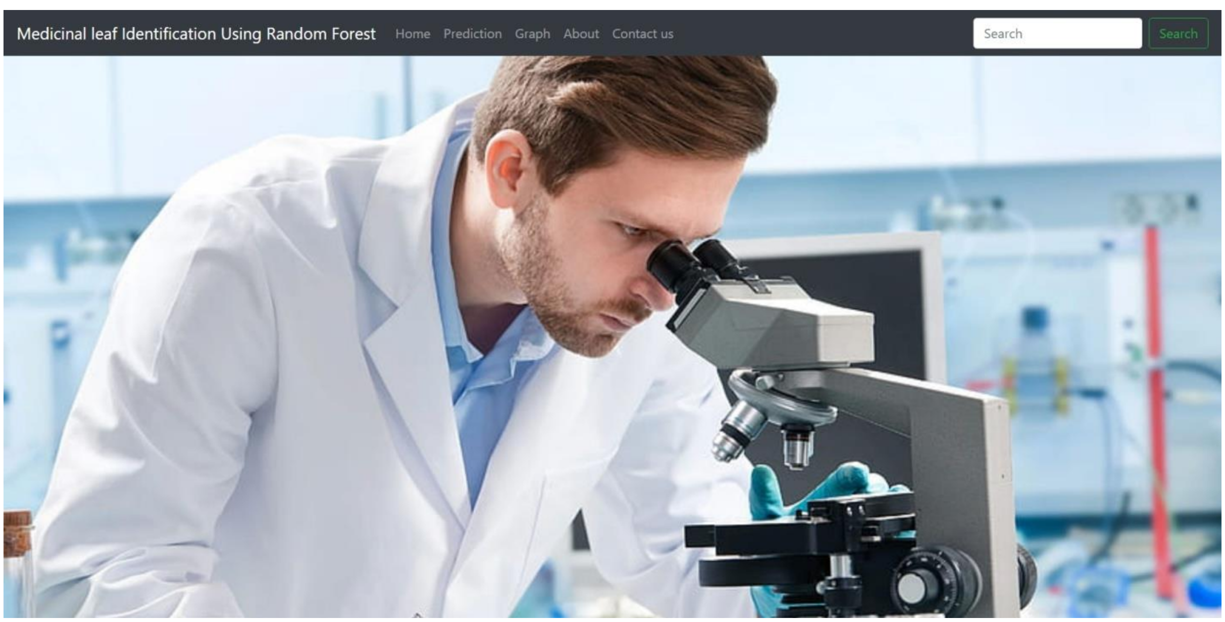

Figure 3: Home page

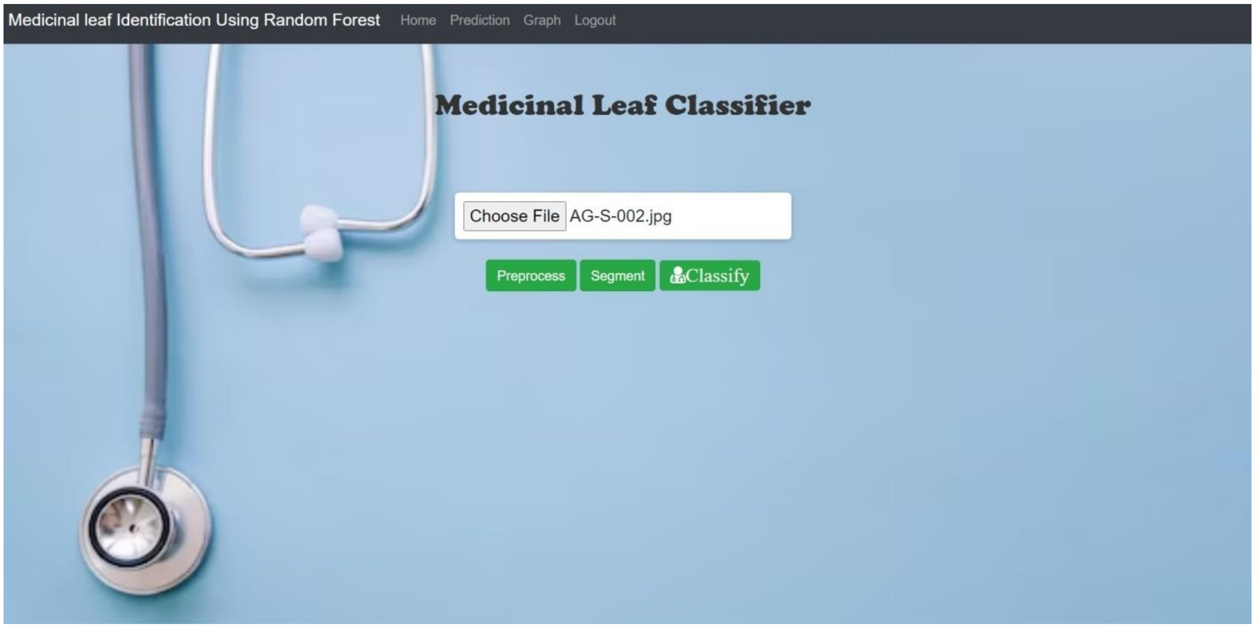

Figure 4: Upload an image

Main window of the medicinal leaf classifier where we have several options like choose file, image preprocess, image segment, and image classify.

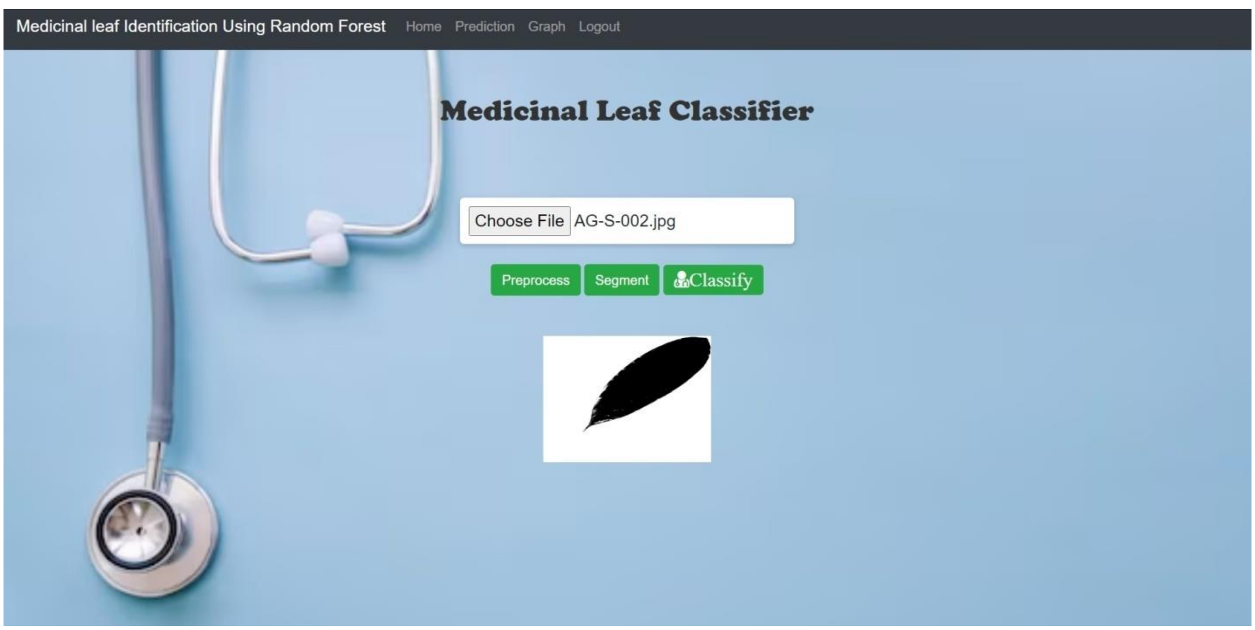

Figure5. Image Preprocessed

The above figure describe the image preprocess. Once we choose file image & next preprocess image to enhance its quality and reduce noise.

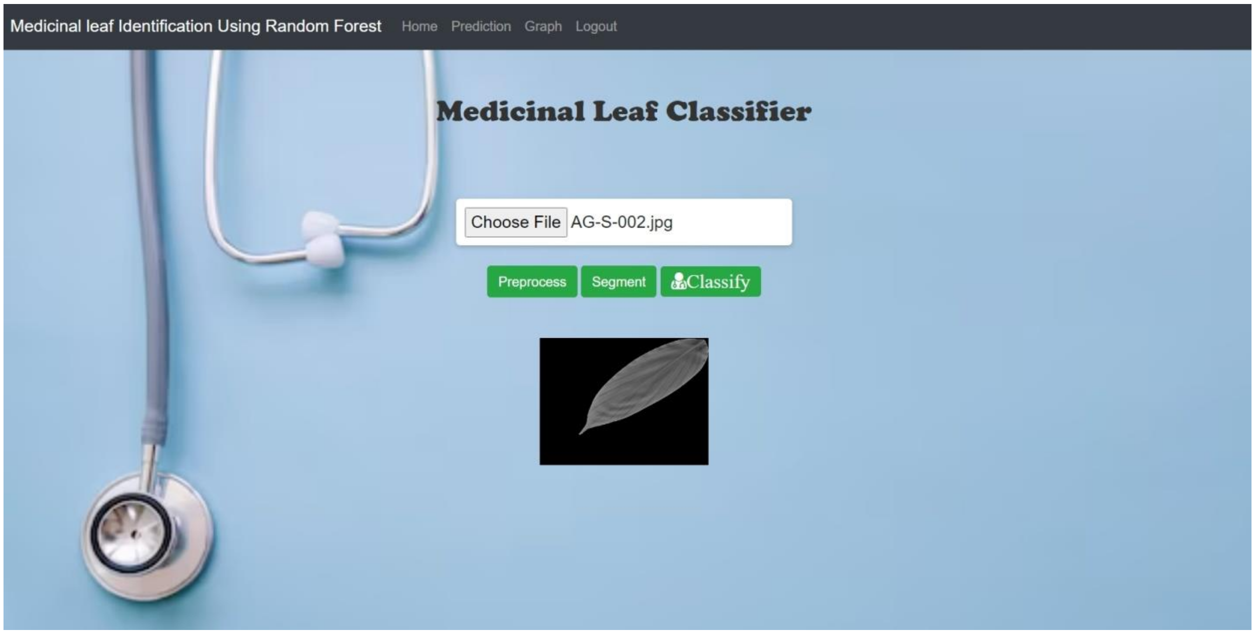

Figure6. Image segmentation

The above figure describe the image segmentation. Once we choose file the image and next preprocess the image next is to image segmentation(segment) to separate the leaf from circumstantial & isolating the relevant portion for analysis.

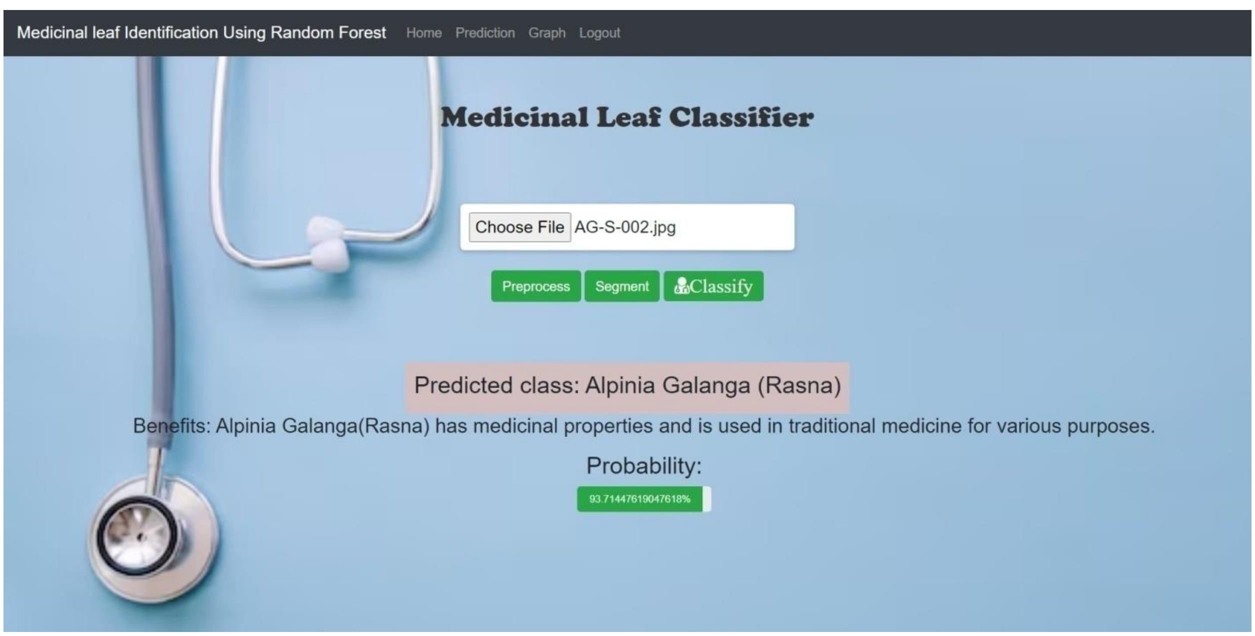

Figure 7: Image classification

The above figure describe the image classification. Which means the final predicted result with benefits and probability of image. in above image the medicinal leaf is classified as Alpinia galangal(rasna).

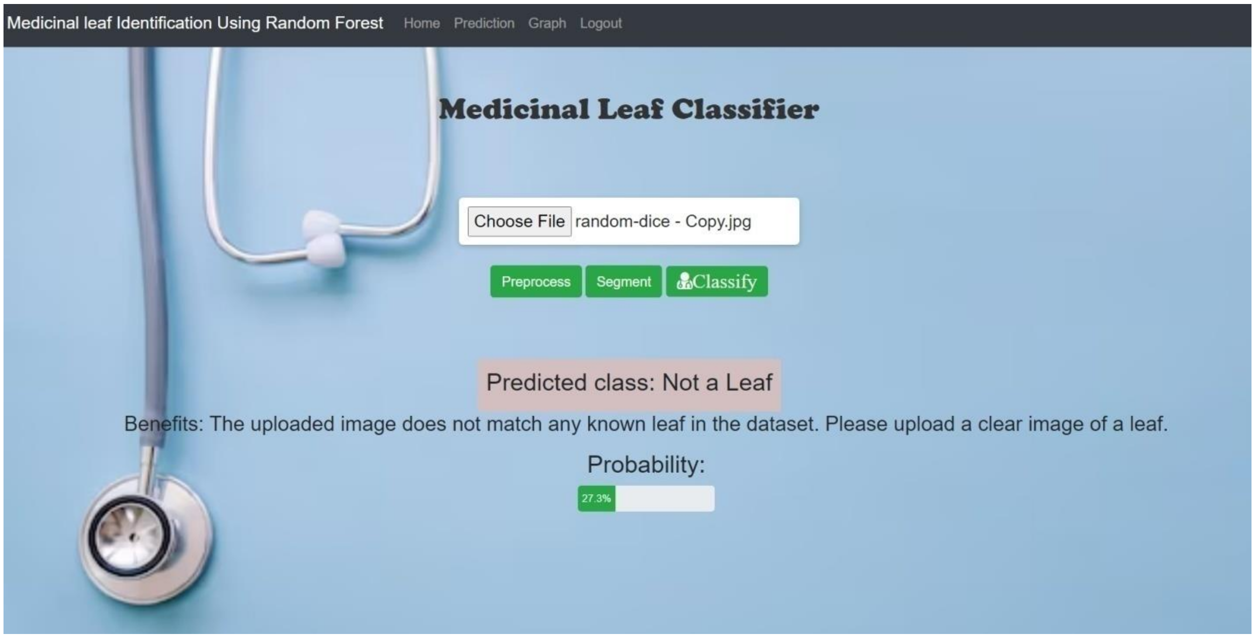

Figure8. Invalid image (not a leaf)

The above figure describe the image classification. When we can choose file of no leaf image or invalid image. in above image the medicinal leaf is classified as not a leaf and the uploaded image dose not match any knoen leaf in the dataset**.**

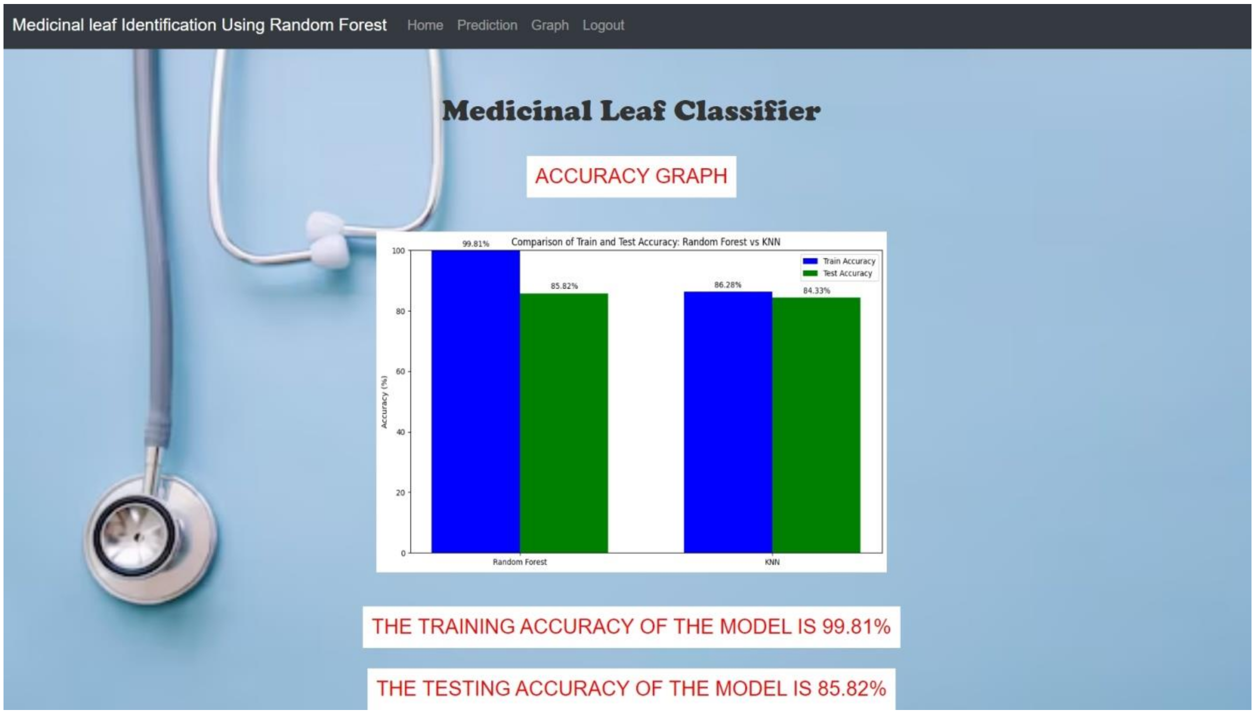

Figure9. Comparison of train & test accuracy: random forest and knn

The above figure describe the accuracy comparison graph of random forest and KNN

VI. SOFTWARE TESTING

6.1 TESTING STRATEGIES

Testing strategies located essential toward ensure reliability, functionality, & performance of software classification. Proposed system employs multi-faceted testing strategy to comprehensively evaluate all components and their interactions. The strategy includes both manual and automated testing methods. Manual testing involves exploratory testing, where testers interrelate by system toward identify unexpected behavior or usability issues. Automated testing, on additional hand, involves writing scripts to automatically implement test cases, confirming that the system behaves as expected under various conditions. Testing approach also includes regression testing, which confirms that innovative code vicissitudes do not unpleasantly affect prevailing functionality. Additionally, the strategy incorporates appearance testing toward evaluate system's responsiveness & stability under load. By employing amalgamation of these testing strategies, the project aims to deliver a robust and reliable application that meets all specified requirements.

6.2 LEVELS OF TESTING

6.2.1 UNIT TESTING

Testing focuses on verifying the functionality of individual components or items of software, such as functions or methods. This level of testing is usually achieved by developers during the coding phase. Unit tests are designed to verify that each module accomplishes its envisioned function correctly and handles edge cases gracefully. By isolating each unit, developers cannister rapidly recognize & fix defects, foremost toward supplementary robust code. In the setting of anticipated system, unit testing would involve validating accurateness of sentiment analysis functions, presentation of ML algorithms, & the correct handling of user inputs. Automated unit testing frameworks, such as pytest for Python, are secondhand toward streamline this process, consenting intended effectual & repeatable testing.

6.2.2 INTEGRATION TESTING

Integration testing scrutinizes exchanges amongst integrated units to verify that they exertion organized as expected. This level of testing detects issues that might ascend when individual components are combined, such as data mismatches or interface errors. In anticipated system, integration testing involves verifying the correct collaboration between sentiment analysis module, ML models, & web application. For example, it tests whether the system correctly processes user inputs, performs sentiment analysis, and returns accurate predictions. Integration tests also check the communication between the front-end and back-end components, ensuring seamless data flow and functionality.

6.2.3 SYSTEM TESTING

System testing appraises plenty & unified software structure to certify it encounters quantified necessities. This level of testing comprises difficult scheme as entire, rather than individual components. It includes purposeful testing to authenticate that all features work as intended & non-functional testing to assess recital, sanctuary, & serviceability. In the proposed system, system testing would involve set-ups where users intermingle with the web application to input data, select algorithms, and view predictions. The testing would ensure that the system handles various use cases, performs efficiently under load, and maintains security standards. System testing helps validate the overall behaviour of the package and ensures it distributes probable outcomes.

6.2.4 VALIDATION

Validation testing certifies that package encounters manipulator's desires & necessities. It comprises gaging system's functionality against corporate necessities & checking whether package fulfills its intended purpose. Validation is often performed through user getting testing ,where end-users test the system in a real-world environment. For the proposed system, validation testing comprises congregation criticism since fiscal analysts, investors, and other users to ensure the application meets their expectations for stock price prediction and sentiment analysis. This difficult segment aids detect slightly gaps between the developed system and user requirements, allowing for adjustments before the final deployment.

6.2.5 OUTPUT TESTING

Output testing authenticates that software produces the correct outputs grounded on numerous inputs and scenarios. This level of testing guarantees that the data presented to users, such as stock price predictions and sentiment scores, are accurate and reliable. In the proposed system, output testing involves comparing the system's predictions against historical data and known outcomes to validate their accuracy. It also includes checking the format, readability, and exposition of outputs on web interface.

6.3 TEST CASES

Table 3: Test Cases

Test Case ID	Test Case Description	Expected Result	Actual Result	Pass/Fail
TC-01	Upload an images of an Alpinia Galanga (Rasna)	The app Should correctly classify the image as Alpinia Galanga (Rasna)	The app Should correctly classifies the image as Alpinia Galanga (Rasna)	Pass
TC-02	Upload an images of an Amaranthus Viridis (Arive- Dantu)	The app Should correctly classify the image as Amaranthus Viridis (Arive-Dantu)	The app Should correctly classifies the image as Amaranthus Viridis (Arive-Dantu)	Pass
TC-03	Upload an images of an Azadirachta Indica (Neem)	The app Should correctly classify the image as Azadirachta Indica (Neem)	The app Should correctly classifies the image as Azadirachta Indica (Neem)	Pass
TC-04	Upload an images of an Basella Alba (Basale)	The app Should correctly classify the image as Basella Alba (Basale)	The app Should correctly classifies the image as Basella Alba (Basale)	Pass
TC-05	Upload an images of an Carissa Carandas (Karanda)	The app Should correctly classify the image as Carissa Carandas (Karanda)	The app Should correctly classifies the image as Carissa Carandas (Karanda)	Pass
TC-06	Upload an images of an Citrus Limon (Lemon)	The app Should correctly classify the image as Citrus Limon (Lemon)	The app Should correctly classifies the image as Citrus Limon (Lemon)	Pass
TC-07	Upload an images of an \ Ficus Auriculata (Roxburgh fig)	The app Should correctly classify the image as Ficus Auriculata (Roxburgh fig)	The app Should correctly classifies the image as Ficus Auriculata (Roxburgh fig)	Pass
TC-08	Uplo Ficus Religiosa (Peepal Tree)ad an images of an	The app Should correctly classify the image as Ficus	The app Should correctly classifies the image as Ficus	Pass

		Religiosa (Peepal Tree)	Religiosa (Peepal Tree)	

Test Cases Table3: This table lists the specific test cases designed to assess exactness of leaf identification system. Each entry includes a unique test case ID, the input image (described by its associated medicinal plant label), and the expected label that the system should return. The persistence of this table is to confirm that scale of medicinal plants is tested to validate the system's performance across diverse forms of leaves.

6.4 TEST RESULTS

Table 4: Test Results

Test Case ID	**Actual Label**	**Status**
TC-01	Alpinia Galanga (Rasna)	Pass
TC-02	Amaranthus Viridis (Arive-Dantu)	Pass
TC-03	Azadirachta Indica (Neem)	Pass
TC-04	Basella Alba (Basale)	Pass
TC-05	Carissa Carandas (Karanda)	Pass
TC-06	Citrus Limon (Lemon)	Pass
TC-07	Ficus Auriculata (Roxburgh fig)	Pass
TC-08	Ficus Religiosa (Peepal Tree)	Pass

Test Results Table4: This table presents the actual outcomes of the test cases. For each test case ID, it shows the label produced by scheme and grade of the test, indicating whether the result matched the expected label. All test cases in this example passed, demonstrating that the system correctly identified each medicinal leaf conferring to specified labels. This result indicates a high accuracy of model in classifying the medicinal plant images.

VII. CONCLUSION

This project aimed to bridge the gap between traditional botanical knowledge and modern technology through amalgamation of ML with medicinal leaves. The objectives were threefold: to invent a dataset, harness control of the Random Forest algorithm, and develop an interactive web application. Reflecting on the accomplishments achieved and insights gained, several key outcomes stand out.The compilation of a diverse Medicinal Leaves Dataset served as the foundation of these efforts. This repository of high-quality leaf images, along with associated medicinal properties, not only fueled the training of model but also stands as witness to amalgamation of tradition and innovation.An intuitive Flask-based web application was created, allowing users to upload leaf images, invoke the model's capabilities to identify leaves, and provide insights into their medicinal attributes. This application embodies the project's essence—making the wealth of botanical knowledge accessible to everyone, regardless of their background.This project serves as a stepping stone toward a more integrated, accessible, and informed exploration of the natural world's healing potential.

8. FUTURE ENHANCEMENT

Looking ahead, this project lays substance intended several compelling future directions. Exploring innovative ML procedures, such as deep learning demonstrates like CNNs , could further enhance the classification accuracy. The dataset's expansion with a wider range of plant species and their medicinal attributes will bolster the model's capabilities. Extending the web application to a mobile platform would make on-the-go leaf identification and information retrieval possible. Multilingual support and user contributions could increase inclusivity and enrich the dataset. Moreover, collaborations with botanical institutions, integration of augmented reality, educational modules, disease detection, and partnerships with conservation efforts offer pathways for growth.

REFERENCES

1. Priyanka Kulkarni, & Dr. Swaroopa Shastri. (2024). Rice Leaf Diseases Detection Using Machine Learning. Journal of Scientific Research and Technology, 2(1), 17–22. https://doi.org/10.61808/jsrt81
2. Shilpa Patil. (2023). Security for Electronic Health Record Based on Attribute using Block-Chain Technology. Journal of Scientific Research and Technology, 1(6), 145–155. https://doi.org/10.5281/zenodo.8330325
3. Mohammed Maaz, Md Akif Ahmed, Md Maqsood, & Dr Shridevi Soma. (2023). Development Of Service Deployment Models In Private Cloud. Journal of Scientific Research and Technology, 1(9), 1–12. https://doi.org/10.61808/jsrt74
4. Antariksh Sharma, Prof. Vibhakar Mansotra, & Kuljeet Singh. (2023). Detection of Mirai Botnet Attacks on IoT devices Using Deep Learning. Journal of Scientific Research and Technology, 1(6), 174–187.
5. Dr. Megha Rani Raigonda, & Shweta. (2024). Signature Verification System Using SSIM In Image Processing. Journal of Scientific Research and Technology, 2(1), 5–11. https://doi.org/10.61808/jsrt79
6. Shri Udayshankar B, Veeraj R Singh, Sampras P, & Aryan Dhage. (2023). Fake Job Post Prediction Using Data Mining. Journal of Scientific Research and Technology, 1(2), 39–47.
7. Gaurav Prajapati, Avinash, Lav Kumar, & Smt. Rekha S Patil. (2023). Road Accident Prediction Using Machine Learning. Journal of Scientific Research and Technology, 1(2), 48–59.
8. Dr. Rekha Patil, Vidya Kumar Katrabad, Mahantappa, & Sunil Kumar. (2023). Image Classification Using CNN Model Based on Deep Learning. Journal of Scientific Research and Technology, 1(2), 60–71.
9. Ambresh Bhadrashetty, & Surekha Patil. (2024). Movie Success and Rating Prediction Using Data Mining. Journal of Scientific Research and Technology, 2(1), 1–4. https://doi.org/10.61808/jsrt78
10. Dr. Megha Rani Raigonda, & Shweta. (2024). Signature Verification System Using SSIM In Image Processing. Journal of Scientific Research and Technology, 2(1), 5–11. https://doi.org/10.61808/jsrt79
11. Dr. Megha Rani Raigonda, & Shweta. (2024). Signature Verification System Using SSIM In Image Processing. *Journal of Scientific Research and Technology*, *2*(1), 5–11. https://doi.org/10.61808/jsrt79
12. Jyoti, & Swaroopa Shastri. (2024). Gesture Identification Model In Traditional Indian Performing Arts By Employing Image Processing Techniques. *Journal of Scientific Research and Technology*, *2*(3), 29–33. https://doi.org/10.61808/jsrt89
13. M Manoj Das, & Dr. Swaroopa Shastri. (2025). Machine Learning Approaches for Early Brain Stroke Detection Using CNN . Journal of Scientific Research and Technology, 3(6), 243–250. https://doi.org/10.61808/jsrt248
14. Abhishek Ashtikar, & Dr. Swaroopa Shastri. (2025). A CNN Model For Skin Cancer Detection And Classification By Using Image Processing Techniques. *Journal of Scientific Research and Technology*, *3*(6), 251–263. https://doi.org/10.61808/jsrt250
15. Dr. Megha Rani Raigonda, & Anjali. (2025). Identification And Classification of Rice Leaf Disease Using Hybrid Deep Learning. *Journal of Scientific Research and Technology*, *3*(6), 93–101. https://doi.org/10.61808/jsrt231