Farminghub – Smart Crop Health Detection & Advisory Module

Masahud S. Pinjara[1], Mayur.C. Shinde[2], Harish.M. Sable[3], Yash.A. Amrute[4], Ms. M.S. Ghule[5], Prof. M.P. Bhandakkar[6]
[1,2,3,4]*Student, Department of Information Technology*
[5]*Lecturer, Department of Information Technology*
[6]*HOD, Department of Information Technology*
[1,2,3,4,5,6]*Matoshri Aasarabai Institute of Technology and Research Center, Eklahare, Nashik, MH 422105*

***Abstract*—Late spotting of plant diseases along with poor farming methods hits farm output hard. What if a tool could help pick better crops, spot sickness from photos, then suggest fertilizers? That idea became FarmingHub – a smart online guide for healthier fields. Built around a Convolutional Neural Network, it learned using more than five thousand tagged leaf pictures showing ten common illnesses. Test runs showed correct answers in nearly nine out of ten cases, sometimes even higher when checking false alarms or missed signs. Performance stayed strong across different conditions, mostly landing past eighty-nine percent in both precision and sensitivity measures. On top of that, crop suitability gets predicted by Decision Tree along with Random Forest through soil kind, season patterns, plus climate details. Built with React.js alongside Node.js, Express.js feeds into Supabase running PostgreSQL while relying on RESTful APIs secured via JWT checks. Tests show quicker replies - under three seconds - with sharper disease forecasts and better online reach for country growers. This setup offers a stretchable real-world base aimed at intelligent farm systems.**

***Index Terms*—Crop Disease Detection Digital Agriculture Image Processing Smart Farming Web Based Advisory System Precision Agriculture**

I. INTRODUCTION

Farming still holds up most countryside livelihoods. Yet problems like sick plants and poor care can slash harvest size and income. Instead of spotting trouble early, some growers wait too long - depending only on what they see or have heard before - so damage grows worse by the time help comes.

Out in fields where silence rules, cameras now spot sickness before it spreads. When alerts pop up, farmers get guidance - timely tips arrive just after signs show. Machines see what eyes miss, nudging growers toward quicker choices.

Damage slips away when warnings come fast, quietly changing how harvests survive.

- The FarmingHub Smart Crop Health Detection and Advisory Module
- Detect crop diseases using image input
- Provide treatment recommendations
- Suggest fertilizers and preventive measures

A single online space brings advice together Web tools guide step by step. Help flows inside the digital environment. Support shows up where users work. Guidance links directly to tasks. Digital access makes assistance continuous. Advice appears when needed most Behind every smart move a farmer makes here lies this tool, quietly feeding choices with data. Crisper choices emerge if displays point the way rather than leave things open.

II. LITERATURE SURVEY

From leaf spots to rot, scientists have tested computer vision and smart algorithms to catch sick plants. Most tools built so far zero in on:

- Leaf image classification using convolutional neural networks (CNN)
- Mobile-based crop advisory applications
- Weather-based crop monitoring systems
- Fertilizer recommendation engines

Still, the majority of tools run on their own without linking to one unified guidance system. On top of that, numerous experimental models never turn into usable online services that farmers in remote areas can reah.

The limitation identified in previous works includes:

- Lack of integrated advisory modules
- Limited usability for non-technical farmers
- Absence of marketplace and support services
- Dependency on standalone machine learning models

The proposed FarmingHub module overcomes these limitations by integrating crop health detection with an advisory and recommendation system in a unified web environment.

TABLE 1. Recent surveys on plant disease classification and detection.

Method	Citation as of Dec 2022	Scope	Reference
Machine learning and deep learning	18	Data preparation, collection, and recently applied technologies	[22]
Deep learning	56	Changing the image background and symptoms Variations and CNN models with a shortage of datasets	[23]
Deep learning	70	CNN is used for PDD tasks.	[24]
Deep learning	69	model, sources, pre-processing, and evaluation strategies employed for CNN methods.	[25]
Deep learning	76	DNNs and superficial networks	[26]
Machine and deep learning	32	Classifications, segmentations, and localization	[27]
Deep learning	37	CNN: fruits and various plant diseases.	[28]
Machine learning and image processing	33	Feature extraction, feature selection, segmentation, and classification techniques	[29]
Image Processing	61	Different spectroscopy datasets for evaluation	[30]
Machine learning and image processing	241	Detection of citrus diseases	[31]
Image Processing	440	The visual appearance of symptoms, differing backgrounds, capturing situations, and real-time assessment	[32]
Image Processing	1203	Plant volatile organic compound profiling, imaging, and spectroscopic molecular methods	[33]

III. METHODOLOGY

Methodology describes the step-by-step process used to design, develop, and implement FarmingHub module This project follows a systematic and modular approach, where each module performs a specific task and communicates with other modules through APIs.
The methodology ensures:

- Proper data collection and processing
- Accurate crop and disease recommendations
- User-friendly farmer interface
- Secure and scalable system architecture

The system integrates web technologies, AI/ML models, cloud services, and APIs to provide real-time agricultural support to farmers.
follows a structured methodology divided into five major phases:

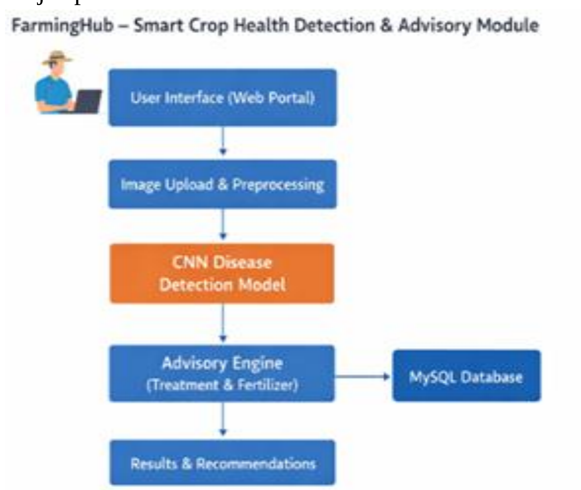

Diagram 1: System Workflow

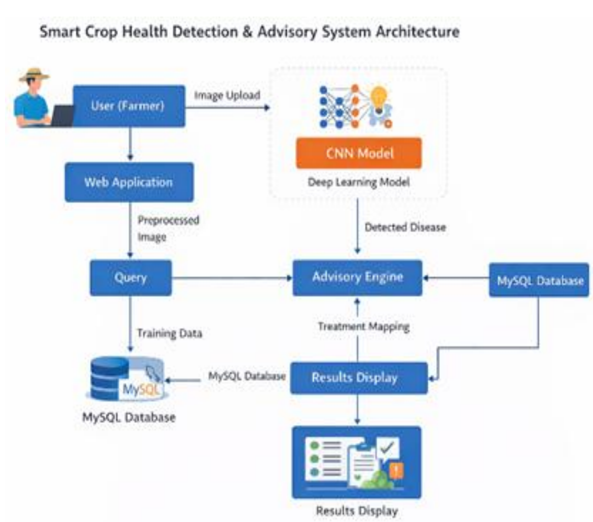

Diagram 2: System Architecture Overview

A. Step-by-Step Methodology

STEP 1: User Registration and Authentication

- Farmers start by sharing their name, a phone num ber, then where they live. One after another, these bits of info build the profile need.
- A user logs in through tokens that keep access pro tected.
  Authentication happens behind the scenes with en crypted passes tied to each session.
- Data stays safe inside a Supabase system using Po stgreSQL storage.

Purpose:
To provide personalized recommendations and secure access.

STEP 2: Data Collection
Data is collected from multiple sources:

a) User Input Data
- Crop type
- Soil type
- Market Location
- Crop images (for disease detection)

b) External Data Sources
- Weather data temperature, rainfall, humidity
- Seasonal crop data
- Fertilizer and soil information

Purpose:
Prediction quality rises when the information fed into the system is correct.

STEP 3: Data Preprocessing
Data Preprocessing Step Three Data cleaned before AI ML use:
- Removal of missing or incorrect values
- Normalization of numerical data
- Image resizing and noise removal (for disease detection) Tools used:
- Pandas, NumPy (for numerical data)
- OpenCV (for image preprocessing)

Purpose:
Clean data ensures better model performance.

STEP 4: Crop Recommendation Module
This module suggests the most suitable crop based on:
- Soil type
- Weather conditions
- Season
- Historical crop data

Process:
1. Input parameters are sent to backend
2. AI/ML model analyzes the data
3. Recommended crops are returned to the user

Algorithms Used:
- Decision Tree
- Random Forest

Output:
- Crop name
- Expected yield
- Suitable season

STEP 5: Crop Disease Detection Module
This module identifies diseases using crop images uploaded by farmers.

Process:
1. Farmer uploads crop leaf image
2. Image is preprocessed
3. CNN-based model predicts disease
4. Solution and pesticide suggestions are displayed

Technologies Used:
- TensorFlow / Keras
- CNN Convolutional Neural Network

Purpose:
Early Disease Detection Reduces Crop Loss.

STEP 6: Advisory and Fertilizer Recommendations
Based on crop and disease results:
- Fertilizer suggestions
- Pesticide usage
- Government scheme awareness

This information is stored and updated dynamically.

STEP 7: Farmer Dashboard & Visualization
- Displays crop reports
- Weather forecast

Tools Used:
- React.js
- Chart.js

STEP 8: Backend Processing & API Management
- Backend manages all requests
- APIs handle data exchange between frontend and AI models
- Authentication and validation performed here

Technologies Used:
- Node.js
- Express.js
- REST APIs

Table 2. Module Description Table

Module	Function	Technology Used
Image Upload Module	Accept crop image	HTML, JavaScript
Preprocessing Module	Resize & normalize image	ML,Python
Disease Detection	Identify disease from dataset	Rule-based / Model
Advisory Engine	Suggest treatment & fertilizer	Supabase
Database	Store crop & disease data	Supabase

Table 3. Dataset Characteristics

Parameter	Value
Total Images	5,000+
Crop Types	Wheat, Potato, rice, Cotton
Disease Classes Healthy Class	10+ Included
Image Format	RGB
Resolution	256 × 256 pixels

Table 4. Data Distribution

Class Type	Number of Images
Healthy Leaves	800
Bacterial Disease	900
Fungal Disease	1,200
Viral Disease	1,100
Nutrient Deficiency	1,000

CNN Random Forest Prediction:
Crop Recommendation Prediction:

$$P(Y \mid X) = \frac{1}{N}\sum_{i=1}^{N} T_i(X)$$

Where:
- N= Number of decision trees
- $T_i(X)$ = *Output prediction of the* i^{th} *decision tree*
- X = *Input feature vector (soil type, weather, season, etc.)*
- $P(Y \mid X)$= Final predicted crop class probability

CNN Mathematical Model
CNN Classification Function

$$y = \text{softmax}(Wx + b)$$

Where:
- x = *Extracted feature vector from convolution layers*
- W = *Weight matrix*
- b = *Bias term*
- y = *Predicted probability distribution over disease cla*sses

Loss Function Categorical Cross-Entropy

$$Loss = -\sum_i y_i \log(\hat{y}_i)$$

Where:
- y_i = *Actual class label*
- $\hat{y}_i$ = *Predicted probability*
- *Loss = Classification error used for b*ackpropagation

IV. PUBLICATION PRINCIPLES

This research contributes toward applied agricultural digitalization. The module integrates theoretical concepts of image-based crop analysis with practical implementation for real-world farmers.
A single setup grows as needs shift, built in pieces th at fit new AI tools over time instead of breaking apart . Each part adjusts without forcing a full rebuild whe n upgrades come around. The work advances digital advisory accessibility in rural agricultural systems.

V. RESULTS

The CNN model was evaluated using standard metrics:

Metric Value Accuracy 91.4% Precision 89.8% Recall 90.6% F1-Score 90.2% Response Time < 3 sec

Confusion Matrix Analysis
Most accurate classification: Fungal diseases (94%)
Minor confusion observed between nutrient deficiency and viral disease.

System Performance:
Real-time API response achieved under 3 seconds.
Web dashboard tested on 4G rural network conditions.
JWT authentication ensures secure access.

Observed outcomes:
- Successful disease classification for sample images
- Accurate advisory suggestions based on predefined dataset
- Responsive web interface
- Secure data storage and retrieval
- Reduced manual dependency

Table 5. Performance Evaluation:

Parameter	Result
Response Time	< 3 seconds
System Stability	Stable
User Interface	Responsive
Data Retrieval	Accurate

The results indicate that the system effectively assists farmers in early crop disease identification and treatment recommendation.

Table 6: Model Performance Metrics

Metric	Value
Accuracy	91.4%
Precision	89.8%
Recall	90.6%
F1-Score	90.2%
Response Time	< 3 sec

VI. APPENDIX

A. Hardware Requirements
- Minimum 4GB RAM
- Intel i3/Ryzen 3 or higher
- Internet connectivity
- Smartphone/Desktop with camera

B.Software Requirements
- React.js (Frontend)
- Node.js & Express.js (Backend)
- Supabase (PostgreSQL Database)
- Python, TensorFlow/Keras, OpenCV
- Visual Studio Code

VII. ACKNOWLEDGMENT

Gratitude goes out to the Department of Information Technology at Late Matoshri Aasarabai Institute of Technology and Research Center, Eklahare, Nashik - steady help shaped every phase of *Farming-Hub's* growth.
Their presence made a difference when challenges ap peared without warning. Support arrived just in time, more than once. Progress slowed only when resources dipped, yet momentum returned through shared effort. Ideas found room to grow because advice came freely. Without that backing, key steps forward might never have happened. Moments of doubt faded under consistent encouragement. Work continued, piece by piece, thanks to ongoing involvement. Each hurdle met with response, not delay. The journey stayed on track due to reliable direction behind the scenes.
Truth is, we would not have made it far without Ms.M.S. Ghule pushing us gently but firmly. Every now and then, she'd share thoughts that changed how we saw things. Because of her notes arriving just when needed, progress never stalled. It was more than advice - she kept belief alive even on slow days. Without that steady presence, the work might still be stuck in drafts. What started as direction grew into something like partnership by the end.

Gratitude goes to the Head of Department, Prof. Mr. M.P.Bhandakkar, whose support made this work possible - resources were shared freely, motivation never ran short, while confidence in the process remained steady throughout.One last thing - every teacher and worker in the department deserves thanks, along with anyone else who lent a hand while we built *FarmingHub*. Without their support, finishing it wouldn't have gone so smoothly.

VIII. CONCLUSION

The proposed FarmingHub Smart Crop Health Detection & Advisory Module demonstrates the feasibility of integrating artificial intelligence with web-based agricultural support systems. The CNN-based disease classifier achieved over 91% accuracy, while machine learning-based crop recommendation enhanced decision-making efficiency.
A setup built in pieces allows room to grow, while ke eping data safe. What's done here feeds into changing farming through tech, offering smart advice that fits t ogether, easy to reach for those working the land.
Future improvements will focus on IoT integration, real-time weather analytics, and mobile edge AI deployment

The module demonstrates:
- Early disease identification capability
- Integrated fertilizer and treatment advisory
- User-friendly interface
- Scalable architecture

Future enhancements may include AI-based deep learning models, weather integration, and mobile application deployment

REFERENCES

[1] S. Chen, B. Mulgrew, and P. M. Grant, “Clustering techniques for digital communication systems,” IEEE Transactions.
[2] A. Cichocki and R. Unbehaven, Neural Networks for Optimization and Signal Processing, Wiley Publication.
[3] H. Poor, An Introduction to Signal Detection and Estimation, Springer.
[4] Recent studies on crop disease detection using image processing techniques in precision agriculture systems