IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, October 2026
Impact Factor: 8.2

# KrishiTech: A Multilingual Smart Farming Assistance Platform

**Kannika[1], B. K Swetha[2], Bhumika K[3], Drakshayini U[4], Prof. Vinay Kumar[5], Dr. B Sreepathi[6]**
Students, Department of Information Science & Engineering[1,2,3,4]
Faculty, Department of Information Science & Engineering[5]
Professor and HOD, Department of Information Science & Engineering[6]
Rao Bahadur Y. Mahabaleswarappa Engineering College, Ballari, Karnataka, India
Kannikaraichur.ise.rymec@gmail.com,swethaswetha7996@gmail.com,usbhumika@gmail.com,
drakshayaniu103@gmail.com

**Abstract:** *Farmers frequently need to consult separate sources for weather information, crop selection, crop health guidance, market prices, government support, and equipment access. KrishiTech is a web-based farming assistance platform designed to bring these services together in a single interface. It is implemented using the MERN stack: MongoDB, Express.js, React.js, and Node.js. The application includes weather information, rule-based crop recommendations, image-based crop disease screening through an external Plant.id service, agricultural market-price comparison, a farm-profit calculator, fertilizer guidance, government-scheme information, equipment rental and booking, and a seeds-and-fertilizers marketplace. English, Kannada, and Telugu interface options are provided, alongside a multilingual chatbot and browser-based voice navigation. Farmer and administrator functions are separated through role-based access control; account passwords are hashed using bcryptjs, and protected requests use JSON Web Tokens (JWT). This paper presents the platform's requirements, architecture, modules, data model, implementation approach, and preliminary functional observations. The current evidence describes implemented features rather than measured agricultural outcomes. No controlled farmer study, diagnostic benchmark, or yield-impact experiment is claimed. Formal usability evaluation, independent validation of disease predictions, and verification of the timeliness and accuracy of market data remain important next steps*

**Keywords**: Smart agriculture, agricultural decision support, MERN stack, crop disease screening, multilingual interface, voice navigation, digital farming.

## I. INTRODUCTION

Agriculture is affected by weather variability, crop pests and diseases, changing market prices, input costs, and differences in access to technical advice. Digital tools can help make relevant information easier to access, but farmers may need to move between multiple services to obtain it. Language and interface complexity can create additional barriers, particularly when digital services are designed around text-heavy workflows. Research has demonstrated the technical potential of image-based plant disease classification [1], while work on open plant-image repositories has supported the development of mobile crop-health tools [2]. Browser technologies also make speech input possible, although support differs by browser and recognition service [3].

KrishiTech was developed as a consolidated web application intended to make common farming tools available from a single dashboard. Its target users include farmers in Karnataka and Telangana, with English, Kannada, and Telugu language options. Rather than claiming to replace agricultural officers, agronomists, or official government services, the platform is intended to provide convenient first-line information and digital workflows.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, October 2026 Impact Factor: 8.2

The main contributions of this project are: (1) integrating multiple farming-related modules into one MERN-stack application; (2) combining weather information, rule-based crop recommendations, and external-service-based disease screening with market and farm-management utilities; (3) providing multilingual interface text, chatbot interaction.

## II. PROBLEM STATEMENT

The project objectives are to provide a centralized access point for farming information; support preliminary crop selection using soil, season, and location inputs; allow users to submit plant images for disease screening; present weather and market-price information; help farmers estimate revenue, costs, and net profit; explain selected fertilizer and government-scheme information; enable equipment rental and product ordering; and improve accessibility through English, Kannada, Telugu, and voice-based controls.

The present scope is a web application and its implemented workflows. It is not a validated agronomic recommendation engine, a certified plant-disease diagnostic device, a payment processor, or an official source of government eligibility decisions. Outputs should be treated as decision-support information and checked against trusted local agricultural guidance.

## III. LITERATURE SURVEY

Agriculture plays an important role in food production and farmers' livelihoods. Digital technologies help farmers access useful information and improve farming decisions. KrishiTech – Modern Farming Assistant is a MERN-stack web application designed to provide multiple agricultural services through a single platform.

The system includes weather updates, crop recommendations, disease screening, market price comparison, fertilizer guidance, government schemes, profit calculation, equipment rental, and a farming products marketplace. It also supports multilingual interaction and voice navigation in English, Kannada, and Telugu.

KrishiTech uses React.js, Node.js, Express.js, and MongoDB, along with external APIs for weather information and crop disease screening. This chapter reviews related research on digital agriculture and plant disease detection, identifies existing system limitations, and explains the need for an integrated farming assistance platform.

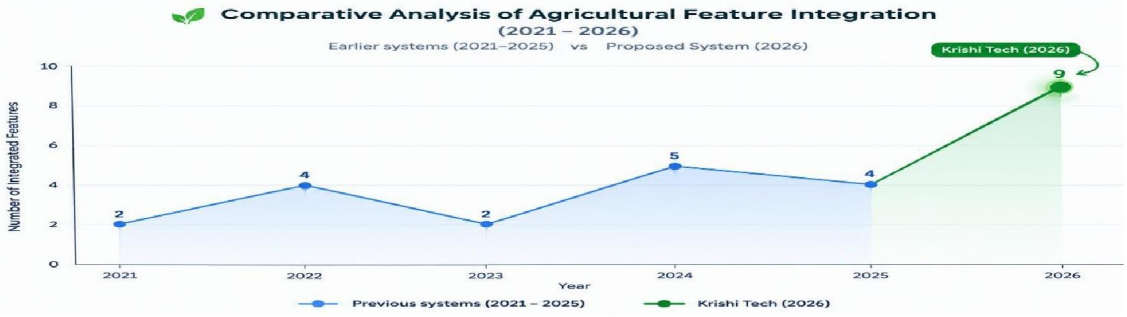
Research on machine learning techniques for plant disease detection explored image-processing methods to identify crop diseases (2022). However, most studies focused mainly on disease detection rather than integrating multiple farming services.

Studies on deep learning and smart agriculture explored automated crop monitoring and disease identification (2023). However, dataset quality, implementation costs, and real-world applicability remained important challenges.

Research on AI-based crop and fertilizer recommendations examined soil and environmental data to support farming decisions (2024). The effectiveness of these approaches depended on input accuracy and model reliability.

Research on computer vision for plant disease detection highlighted challenges such as limited dataset diversity and insufficient real-world field testing (2025).

Recent research trends emphasize integrating intelligent crop recommendations, disease screening, and farmer oriented digital services (2026). However, internet connectivity, data quality, and language accessibility remain important considerations.



IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, October 2026 Impact Factor: 8.2

Existing approaches often focus on individual agricultural functions. KrishiTech aims to integrate weather updates, crop recommendations, disease screening, market prices, profit calculation, fertilizer guidance, government schemes, equipment rental, and a farming marketplace into one multilingual platform.

## IV. METHODOLOGY

The proposed Krishi Tech – Modern Farming Assistant is a web-based agricultural support system designed to provide essential farming services through a single platform. The system follows a modular architecture to ensure easy development, integration, and maintenance. The frontend provides a user-friendly interface for farmers to access different agricultural services, while the backend handles application logic, user requests, and data processing. User registration and login functionality enable access to the supported features. The crop disease detection module analyzes uploaded crop images to identify potential diseases. The weather module provides relevant weather information through the configured data source. The market price module displays available crop price details to support farmers in making informed selling decisions. The fertilizer recommendation module provides fertilizer-related guidance based on the implemented logic and available inputs. The profit calculator estimates expected revenue and profit using the expected yield, selling price, and cultivation costs. A database is used to store and retrieve necessary application information. External APIs are integrated wherever required to obtain updated information from external services. Input validation and error handling help manage invalid data and unavailable services. The individual modules are integrated to provide a unified agricultural assistance platform. Functional and integration testing are conducted to verify the operation of the implemented features. Finally, the system can be evaluated based on functionality, usability, reliability, and measured performance to assess its suitability for agricultural decision support.

## V. IMPLEMENTED FUNCTION MODULES

### Landing Page and Farmer Dashboard

The landing page introduces the service and links to its main capabilities. The farmer dashboard provides navigation to dashboard statistics, weather, disease detection, crop recommendations, market prices, profit calculation, fertilizer guidance, government schemes, equipment rental, orders, marketplace, chatbot, and profile settings. A language selector provides English, Kannada, and Telugu interface options.

### Weather Information

The weather module integrates OpenWeatherMap and presents conditions such as temperature, feels-like temperature, humidity, wind speed, pressure, visibility, sunrise/sunset, and forecast information were returned by the configured service. The interface includes city search and a set of quick-location buttons. The project summary states that a +sevenday forecast, rain alerts, and sowing/harvesting suggestions are included. These suggestions should be treated as general information rather than location-specific agronomic advice unless validated with local agricultural experts. If mock fallback data is enabled when credentials are missing, the interface must clearly label it as simulated and not live weather.

### Crop Recommendation

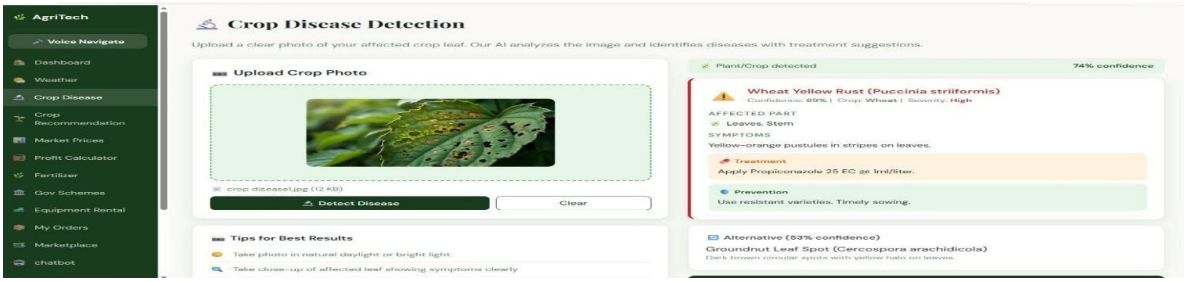
The crop recommendation module accepts soil type, season, and location and returns up to four recommended crops with indicative duration, water requirement, expected yield, and market-price information. The described implementation is rule-based. A rule-based engine is explainable and relatively simple to maintain, but its reliability depends on the rules and source data. The current project summary does not establish that the engine has been evaluated against agronomist decisions or field outcomes.

### Crop Disease Screening

Users can upload JPG, PNG, or WEBP images, with a stated maximum file size of 10 MB. The application validates the file, checks whether the image appears to contain a plant, and sends the image to Plant.id when configured. The result interface can present a healthy/disease indication, candidate disease, confidence, affected part, symptoms, treatment/prevention information, and alternative diagnoses, along with links to local agricultural support. A local

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, October 2026
Impact Factor: 8.2

fallback is described when an API key is unavailable. The fallback must not be presented as equivalent to validated AI diagnosis. Any disease result is preliminary and should be confirmed by a qualified agricultural expert before pesticide or treatment decisions.




**Market Prices and Market Locations**

The market module compares prices for ten crops across five listed markets: Bengaluru, Tumakuru, Mysuru, Hassan, and Hyderabad. The project summary describes MongoDB storage, scheduled updates at 6:00 a.m. Indian Standard Time using node-cron, price-trend indicators, manual refresh, and Recharts-based comparisons. Because a schedule alone does not guarantee that upstream data are current or official, the production system should record the source, market date, unit, last-updated timestamp, and any retrieval failure. Demonstration or seeded prices should be labelled clearly and should not be described as verified live APMC data.

**Profit Calculator and Fertilizer Guide**

The profit calculator accepts crop, area, yield per acre, market price, seed cost, fertilizer cost, labour cost, and other expenses. It calculates estimated total yield, gross revenue, total cost, net profit, and return on investment. These are estimates driven by user inputs, not predictions or guarantees. The fertilizer guide provides crop-specific NPK guidance for rice, wheat, maize, cotton, and groundnut, soil-health tips, and a small product catalogue. Fertilizer recommendations should be checked against soil-test results and region-specific agricultural advice.

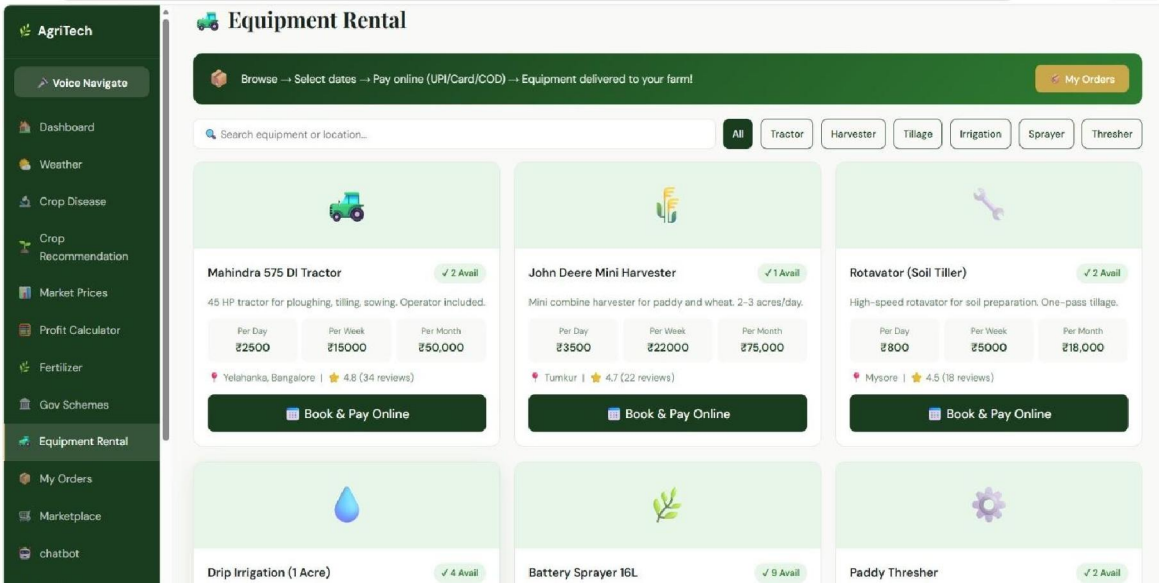
**Government Schemes**

The schemes module summarizes selected programs, including PM-KISAN, Pradhan Mantri Fasal Bima Yojana, Kisan Credit Card, PM Krishi Sinchai Yojana, Soil Health Card, and e-NAM. The interface provides descriptions, eligibility information, document lists, and application links. Scheme rules and links can change, so the module should link to official government sources and display a last-verified date. The listed Kisan Call Centre number should also be rechecked before publication.

**Equipment Rental, Orders, and Marketplace**

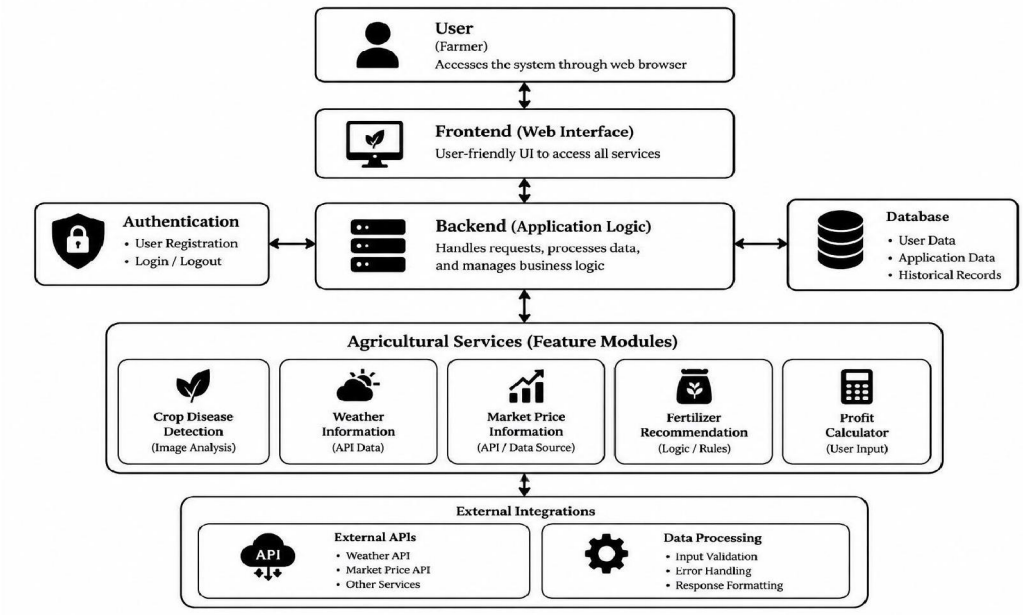
The equipment rental module lists categories such as tractors, harvesters, tillage equipment, irrigation systems, sprayers, and threshers. Users can search/filter listings, select a rental period and rental type, choose a displayed payment method, and submit a booking. Order information is stored in MongoDB, and the application tracks order status and equipment availability. The marketplace lists seed and fertilizer products with category filters, quantity controls, stock indicators, and a cart/order workflow. Unless a payment gateway is actually integrated and independently tested, the listed COD/UPI/card/net-banking options should be described as selected order preferences or interface options—not as proof that secure online payment settlement is operational.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, October 2026
Impact Factor: 8.2




## VI. RESULT

The Krishi Tech – Modern Farming Assistant integrates multiple agricultural services into a single web-based platform. The system provides crop disease detection through uploaded crop images. The weather module displays relevant weather information from the configured data source.




.

The market price module helps farmers access available crop price details. The fertilizer recommendation module provides fertilizer-related guidance based on the implemented logic. The profit calculator estimates revenue and profit using cultivation costs, expected yield, and selling price. The user interface enables convenient access to the available features. Database integration supports the storage and retrieval of application information. API integration helps

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, October 2026
Impact Factor: 8.2

retrieve external data wherever required. The integrated system aims to simplify access to agricultural information and support farmers in making informed decisions. Further testing with actual data is required to measure accuracy, reliability, and overall system performance.

## VII. CONCLUSION

KrishiTech is a MERN-stack web application that brings together weather information, rule-based crop recommendations, image-based crop disease screening, market-price comparison, profit estimation, fertilizer guidance, government-scheme information, equipment rental, marketplace workflows, multilingual chatbot interaction, and voice navigation. Its integrated design aims to reduce the effort required to access several categories of farming information through separate services. The described implementation includes MongoDB-backed records, external API integrations, and role-separated farmer and administrator workflows. At the same time, feature implementation should not be confused with validated agronomic or economic impact. Reliable production use requires transparent data sourcing, careful handling of external service failures, security testing, agricultural-expert validation, and user-centred evaluation. A structured testing and field study programme is the next step toward determining the platform's practical value for farmers.

## REFERENCES

1. S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using Deep Learning for Image-Based Plant Disease Detection," Frontiers in Plant Science, vol. 7, article 1419, 2016. doi: 10.3389/fpls.2016.01419. Available: https://doi.org/10.3389/fpls.2016.01419
2. D. P. Hughes and M. Salathé, "An Open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics through Machine Learning and Crowdsourcing," arXiv:1511.08060, 2015. Available: https://arxiv.org/abs/1511.08060
3. MDN Web Docs, "Web Speech API," Mozilla. Available: https://developer.mozilla.org/enUS/docs/Web/API/Web_Speech_API (accessed 13 Oct. 2026).
4. OWASP Foundation, "Password Storage Cheat Sheet." Available: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html (accessed 13 Oct. 2026).
5. OWASP Foundation, "REST Security Cheat Sheet." Available: https://cheatsheetseries.owasp.org/cheatsheets/REST_Security_Cheat_Sheet.html (accessed 13 Oct. 2026).
6. React, "React Documentation." Available: https://react.dev/ (accessed 13 Oct. 2026).
7. Express.js, "Express—Node.js Web Application Framework." Available: https://expressjs.com/ (accessed 13 Oct. 2026).
8. MongoDB, "MongoDB Manual." Available: https://www.mongodb.com/docs/manual/ (accessed 13 Oct. 2026).
9. OpenWeather, "Current Weather Data API Documentation." Available: https://openweathermap.org/current (accessed 13 Oct. 2026).
10. Plant.id, "Plant Identification API." Available: https://plant.id/ (accessed 13 Oct. 2026).