

International Journal Research Publication Analysis

Page: 01-06

BHOOMITRA: AN AI-POWERED SMART AGRICULTURE PLATFORM FOR INDIAN FARMERS USING MACHINE LEARNING AND GENERATIVE AI

***Monisha J. S., Moulya G. U., Praveen Kumar H. K., Yashaswini T. M.**

Department of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur,
Karnataka, India.

Article Received: 18 September 2026

Article Revised: 08 October 2026

Published on: 28 October 2026

***Corresponding Author: Monisha J. S.**

Department of Computer Science and Engineering, Kalpataru Institute of
Technology, Tiptur, Karnataka, India.

DOI: <https://doi-doi.org/101555/ijrpa.7840>

ABSTRACT

Small and marginal farmers often need to consult separate sources for crop selection, market prices, government schemes, fertilizer guidance, farm planning and protection against animal intrusion. This paper presents BHOOMITRA, a modular web-based smart agriculture platform that combines generative artificial intelligence, computer vision and digital farm services in a single application. The system is implemented using Flask Blueprints, Jinja2 and Bootstrap, with SQLAlchemy-backed PostgreSQL storage in deployment and SQLite for local development. Groq-hosted LLaMA models support crop recommendations, yield and profit estimation, market and scheme information, a farming roadmap and conversational assistance. A multimodal model supports fertilizer-label analysis, while YOLOv8 and OpenCV provide animal detection with local alarm and Twilio voice/SMS alerts. The platform also includes email OTP authentication, a crop marketplace, labour discovery and paid expert consultation through Razorpay. Structured prompts, JSON-oriented outputs, caching and rule-based fallbacks are used to improve resilience. Functional testing reported end-to-end operation across the modules; observed average response times ranged from 0.9 seconds for per-frame intrusion inference to 6.5 seconds for roadmap generation under the report's test conditions. The implementation demonstrates how several farmer-facing services can be integrated into one responsive platform, while recognising that AI-generated advice and estimated figures require local validation.

KEYWORDS: *Smart agriculture, generative AI, LLaMA, Flask, YOLOv8, crop*

recommendation, APMC, fertilizer analysis, farmer services.

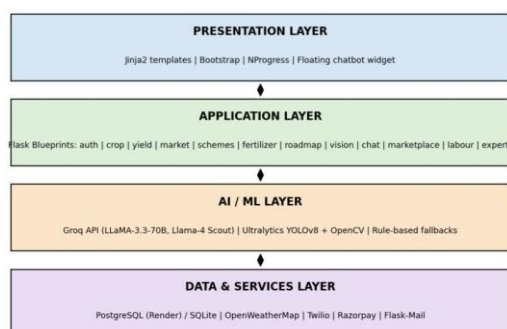


Fig. 1. Layered architecture of BHOOMITRA (adapted from the project report, Fig. 5.1).

I. INTRODUCTION

Agriculture is central to rural livelihoods in India, but farm decisions are made under uncertainty caused by variable rainfall, changing input costs, fluctuating mandi prices and uneven access to trusted advice. Digital agricultural tools can help, yet their functions are often fragmented across independent services. A farmer may need one source for weather, another for prices, another for government support and a separate channel for expert guidance.

BHOOMITRA (meaning “friend of the land”) addresses this fragmentation through a single responsive web platform. Its objective is to make agricultural information and services accessible through a consistent interface, personalised primarily by state, district, soil, season and water availability. Rather than relying on one general-purpose response, each AI-enabled feature uses a task-specific prompt and expected output structure. The platform is intended as decision support; generated estimates are not guarantees and should be checked against local conditions and official sources.

II. SYSTEM OVERVIEW AND ARCHITECTURE

BHOOMITRA follows a layered architecture within a Flask application. The presentation layer uses Jinja2 templates and Bootstrap, including progress indicators and a floating chatbot. The application layer separates features into Flask Blueprints. The AI/ML layer calls Groq-hosted LLaMA models for text and image-related tasks and uses Ultralytics YOLOv8 with OpenCV for animal detection. The data and services layer combines SQLAlchemy, PostgreSQL on Render (SQLite during local development), OpenWeatherMap, email, Twilio and Razorpay.

Fig. 1. Layered architecture of BHOOMITRA (adapted from the project report, Fig. 5.1).

The Blueprint-based design isolates route handling, input validation, prompt construction and persistence by feature. This separation supports independent changes and makes the application easier to test. Structured outputs are parsed before rendering; where an external response is malformed or unavailable, selected modules use cached information or rule-based fallback behaviour.

III. IMPLEMENTATION METHODOLOGY

A. Authentication and Data Management

Registration and login are supported by email one-time-password verification, password hashing and role-aware access. SQLAlchemy models persist user profiles, chat history, yield predictions, alerts, crop listings, expert records and payment-related data. Configuration values and service credentials are supplied through environment variables rather than embedded in application code.

B. Crop Recommendation

The crop recommender accepts contextual inputs such as location, soil type, water availability and season, and incorporates weather information obtained through OpenWeatherMap. A Groq-hosted LLaMA model returns a ranked set of five candidate crops with fields such as suitability score, yield range, water requirement, market demand, estimated profitability and suggested planting month. The application validates and presents the structured response. A rule-based crop list is used when weather or model responses fail.

C. Yield and Profit Prediction

The yield/profit module accepts crop and farm context and asks the language model to return structured estimates for yield, cultivation cost, expected market price, revenue, profit and a confidence indicator. These are estimates generated from model output, not measurements from a calibrated field-trial model; the interface should therefore communicate uncertainty and encourage verification using local agricultural data.

D. Market Price and Government Schemes

The APMC module presents mandi-wise crop price information in a sortable view so farmers can compare markets. The scheme finder uses the farmer's state and district to retrieve potentially relevant Central and State schemes, with official application links requested in the output. Results may be cached to reduce repeated inference calls. Since prices and eligibility rules can change, these services should be treated as time-sensitive and checked against official sources before action.

E. Fertilizer Scanner and Farming Roadmap

The fertilizer scanner accepts an image of a fertilizer label and uses a multimodal LLaMA model to interpret available label information, including NPK composition, dosage guidance and crop suitability. The roadmap generator creates a 12-month crop calendar with activities such as soil preparation, planting and scaling considerations in Markdown format. Image clarity affects label interpretation, while roadmap latency is higher because the response is longer.

F. Animal Intrusion Alerts

The animal intrusion module processes a camera feed using YOLOv8 and OpenCV. When a configured detection condition is met, the system can trigger a local alarm and send voice/SMS notifications through Twilio to the registered farmer. The module depends on camera position, lighting and the target appearing within the detector's capabilities; these environmental factors can affect practical performance.

G. Chatbot, Marketplace, Labour and Experts

The AI chatbot supports multi-turn agricultural queries and stores conversation history. The marketplace supports crop listings and direct interaction between farmers and buyers; the labour portal helps users discover agricultural workers. GOAT Experts provides paid expert consultation, with Razorpay used for payment processing. These services extend BHOOMITRA beyond advice generation into farm-related coordination and transactions.

IV. TESTING AND RESULTS

The project report describes unit, integration, system, compatibility, security, usability and regression testing. All test cases recorded for the final build were executed, and the report states that all ten modules operated end to end. Fault-injection checks using an invalid weather key and malformed JSON showed that the crop module could return a fallback rather than crash. When a scheme API request failed, cached results remained available. These findings are based on project-team testing and do not establish performance at national scale.

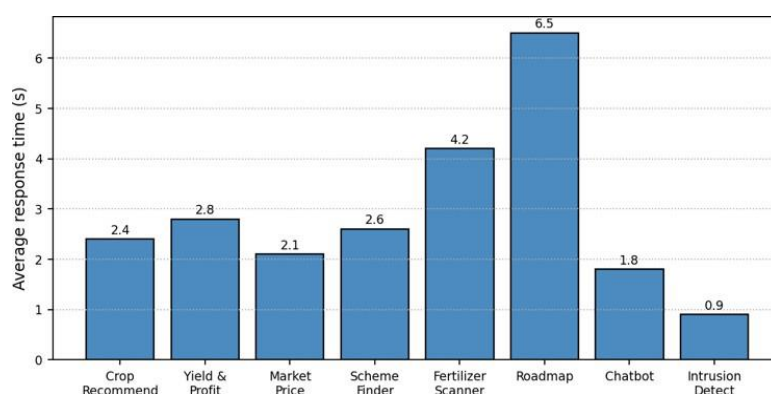


Fig. 2. Average response-time chart reproduced from the project report (Fig. 8.1).

The report records approximate average response times on a typical broadband connection: crop recommendation, 2.4 s; yield and profit, 2.8 s; market price, 2.1 s; scheme finder first search, 2.6 s; fertilizer scanner, 4.2 s; roadmap generator, 6.5 s; chatbot, 1.8 s; and YOLOv8 intrusion inference per frame, 0.9 s. These are indicative observations, not service-level guarantees, and may vary with network conditions, model-provider load and output length.

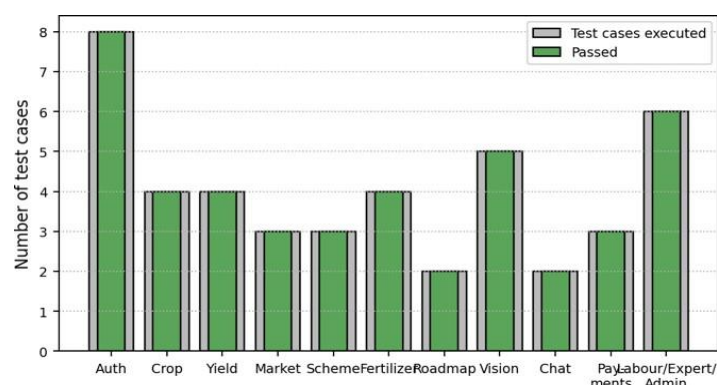


Fig. 3. Test cases executed and passed per module (project report, Fig. 8.2).

V. DISCUSSION

The principal contribution of BHOOMITRA is integration: a farmer can access crop suggestions, economic estimates, market and scheme information, image-based fertilizer interpretation, planning assistance, security alerts and transactional features from one interface. Task-specific prompts and structured outputs help the application convert model responses into predictable interface components. Caching and rule-based fallbacks improve continuity when selected external calls fail.

Several limitations remain. Language-model outputs may contain outdated or inaccurate local information; price and scheme details require current authoritative sources; the project's tests involved a limited set of users and scenarios; and computer-vision performance depends on lighting and camera setup. The report does not provide a large-scale field trial or independently validated agronomic accuracy metrics. Therefore, the platform should be considered a prototype decision-support system rather than a replacement for agricultural officers, agronomists or official records.

VI. CONCLUSION AND FUTURE WORK

This paper presented the implementation of BHOOMITRA, a Flask-based smart agriculture platform that combines generative AI, computer vision, relational data storage and external

communication/payment services. Its modular architecture brings ten farmer-oriented service areas into one responsive application. The project report records successful end-to-end functional testing and provides indicative response-time observations. Future work identified in the report includes stronger integration of verified agricultural and market datasets, broader field testing, improved multilingual access, sensor and satellite-data integration, and additional validation of model outputs under diverse regional conditions.

REFERENCES

1. Meta AI, "The Llama 3 Herd of Models," 2024.
2. Ultralytics, "YOLOv8 Documentation," Ultralytics. [Online].
3. Available: <https://docs.ultralytics.com/>
4. Pallets Projects, "Flask Documentation." [Online]. Available: <https://flask.palletsprojects.com/>
5. Groq, "Groq API Documentation." [Online]. Available: <https://console.groq.com/docs/>
6. OpenWeather, "Current Weather API Documentation." [Online].
7. Available: <https://openweathermap.org/api>
8. Twilio, "Programmable Messaging and Voice Documentation." [Online]. Available: <https://www.twilio.com/docs>
9. Razorpay, "Payment Gateway Documentation." [Online]. Available: <https://razorpay.com/docs/>
10. BHOOMITRA Project Team, "AI-Powered Smart Agriculture Platform for Indian Farmers Using Machine Learning and Generative AI," project report, Department of CSE, Kalpataru Institute of Technology, 2026–2027.