

Krush Sahayak: A Smart Agriculture Web Portal for Farmers

PRATHMESH DATTATRAY KADAM¹, PAVAN KADAM², GAUTAM JOGDANDE³

^{1,2,3}JSPM University, Pune, India

Abstract- Indian agriculture remains the backbone of the national economy, employing approximately 54% of the total workforce and contributing nearly 18% to the country's Gross Domestic Product (GDP). Despite its significance, farmers continue to face critical information gaps: delayed access to government welfare schemes, inability to track real-time commodity prices, and limited awareness of localized weather forecasts. These deficiencies result in suboptimal crop planning, financial losses, and reduced agricultural productivity. This paper presents Krushi Sahayak, a centralized smart agriculture web portal designed to bridge these information gaps. The proposed system integrates a real-time Weather API, live market price feeds, a curated government scheme repository with eligibility filtering, and a push-notification engine within a single, accessible platform. The portal was developed using the Flask micro-framework with a MySQL relational database on the backend, and HTML5/CSS3/JavaScript on the frontend. Usability evaluation with 30 farmers across three districts of Maharashtra demonstrated a System Usability Scale (SUS) score of 79.4, indicating good usability. The portal reduced the average time for a farmer to locate a relevant government scheme from 47 minutes (using existing fragmented sources) to under 3 minutes. These results validate Krushi Sahayak as an effective tool for accelerating digital transformation in agriculture.

Index Terms- Smart Agriculture, Web Portal, Farmers, Weather API, Government Schemes, Market Prices, Digital Farming, Flask, MySQL, Usability.

I. INTRODUCTION

Agriculture is the primary occupation of more than half of India's population, yet it remains one of the most information-deprived sectors. A 2023 survey by the National Sample Survey Office (NSSO) reported that only 22% of farming households had access to any form of digital agricultural advisory. The consequences are tangible: farmers sow crops without reliable weather intelligence, sell produce without knowledge of prevailing mandi prices, and

remain unaware of subsidies or crop insurance schemes to which they are entitled [1].

Existing digital agricultural services are fragmented. The Government of India maintains separate portals for schemes (PM-KISAN portal), weather (IMD website), and mandi prices (Agmarknet). A farmer seeking all three types of information must navigate multiple websites, often designed with urban, literate users in mind. Research has shown that information silos significantly increase the cognitive load on end-users and reduce adoption of digital public services [2].

This paper proposes Krushi Sahayak (literally, "Agricultural Helper" in Marathi), a unified web portal that aggregates weather forecasts, crop market rates, government scheme details, and real-time notifications into a single, mobile-responsive interface optimized for low-bandwidth rural connectivity. The key contributions of this work are: (i) design of an integrated information architecture for agricultural data, (ii) implementation of a scheme eligibility engine that filters schemes based on farmer-supplied demographic parameters, (iii) a lightweight notification subsystem for alerts on price thresholds and weather warnings, and (iv) empirical usability evaluation with actual end-users in field conditions.

II. RELATED WORK

Significant prior work exists at the intersection of information technology and agriculture. Liakos et al. [3] surveyed machine learning applications in precision agriculture, demonstrating the potential of data-driven systems for yield prediction and pest detection. However, such solutions demand sensor infrastructure and connectivity that are unavailable in most small-holding environments in India.

Wolfert et al. [4] examined big data in smart farming and emphasized that value is realized only when information reaches farmers in an actionable format. They noted that real-time dashboards outperform periodic report delivery in farmer decision-making contexts. Consistent with this finding, Krushi Sahayak adopts a live-data dashboard approach over static periodic reports.

Sharma and Jain [5] proposed a mobile application for crop advisory using weather APIs and found a 31% improvement in irrigation scheduling efficiency among pilot users. Their work validates API-driven weather integration as a practical intervention. However, the application did not address government scheme discovery, which our work incorporates.

In the domain of e-governance, Bhatnagar [6] analyzed India's agricultural e-portal landscape and identified fragmentation as the primary adoption barrier, recommending a "single-window" approach. Krushi Sahayak directly addresses this recommendation. Table 1 contextualizes related systems against the features offered by the proposed portal.

Table 1: Comparison of Related Agricultural Information Systems

System / Study	Weather Info	Market Rates	Govt. Schemes	Notifications	Unified Portal
Agmarknet [GoI]	No	Yes	No	No	No
IMD Website [GoI]	Yes	No	No	No	No
PM-KISAN Portal [GoI]	No	No	Yes	Limited	No
Sharma & Jain [5]	Yes	No	No	No	No
Krushi Sahayak (Proposed)	Yes	Yes	Yes	Yes	Yes

III. PROBLEM STATEMENT AND MOTIVATION

The core problem addressed in this work is the absence of a unified, farmer-centric digital interface that delivers real-time, actionable agricultural information within the constraints of rural internet connectivity (2G/3G networks, intermittent availability) and limited digital literacy. Specifically, three pain-points motivated the design of Krushi Sahayak:

- **Scheme Unawareness:** Only 31% of eligible farmers in Maharashtra claimed benefits under the Pradhan Mantri Fasal Bima Yojana (PMFBY) in 2022-23, primarily due to lack of awareness [7].
- **Market Price Opacity:** Without access to current mandi rates, farmers often sell at farm-gate to intermediaries at prices 15-40% below prevailing mandis, losing significant income [8].
- **Weather-Induced Crop Loss:** Unplanned irrigation and sowing schedules due to absence of hyperlocal weather forecasts contribute to an estimated Rs. 50,000 crore annual crop loss nationally [9].

IV. SYSTEM DESIGN AND ARCHITECTURE

4.1 Architectural Overview

Krushi Sahayak follows a three-tier client-server architecture: a presentation tier (HTML5/CSS3/JavaScript frontend), an application logic tier (Python Flask RESTful backend), and a data tier (MySQL relational database). The three tiers communicate via HTTP/HTTPS, with the backend exposing RESTful API endpoints consumed by the frontend. External data is ingested through two third-party APIs: the OpenWeatherMap API for meteorological data and a state government commodity price feed for daily mandi rates.

The system is deployed on a cloud virtual machine (2 vCPU, 4 GB RAM) running Ubuntu 22.04 LTS with Nginx as the reverse proxy and Gunicorn as the WSGI application server. This configuration supports approximately 500 concurrent users before requiring horizontal scaling.

4.2 Module Descriptions

The portal comprises five core functional modules, each encapsulated as a Flask Blueprint to enforce separation of concerns:

- **Weather Module:** On page load, the user's district is resolved from their registered profile. A scheduled background task (APScheduler, every 30 minutes) fetches 5-day forecasts from the OpenWeatherMap One-Call API 3.0 and caches results in a Redis store. Cached data is served to the frontend, ensuring low latency even under API rate limits.
- **Market Rates Module:** Daily mandi prices are ingested at 06:00 IST from the Maharashtra Agri Marketing Board (MSAMB) open data feed. Data is normalized into the MarketPrice table and displayed with 7-day trend sparklines using Chart.js.
- **Government Schemes Module:** A curated database of 47 central and 23 Maharashtra state schemes is maintained. Each scheme record includes eligibility criteria (land holding size, caste category, crop type, income limit). A filtering engine performs SQL-based eligibility matching against the user's stored profile, returning only applicable schemes. Scheme records are updated monthly by an administrator.
- **Notification Module:** Farmers subscribe to alerts for specific commodities or weather thresholds (e.g., "Alert me when tomato price at Pune mandi exceeds Rs. 40/kg" or "Alert me when rainfall forecast exceeds 50mm"). Alerts are delivered via browser push notifications (Web Push Protocol, VAPID authentication) and optional SMS via the Textlocal gateway.
- **Authentication Module:** User registration collects name, mobile number, district, land holding, caste category, and primary crops grown. Passwords are hashed using bcrypt (work factor 12). Sessions are managed via server-side Flask-Login with a 30-day remember-me token stored as a secure, HttpOnly cookie.

4.3 Database Schema

The MySQL database comprises nine tables. Key entities include: User (profile and authentication), Scheme (scheme details and eligibility criteria), MarketPrice (commodity prices per mandi per date), WeatherCache (cached API responses), Alert (user-defined notification triggers), and Notification (delivery log). Referential integrity is enforced via foreign key constraints. Indexes are placed on frequently queried columns (user_id, district_id, commodity_id, price_date) to support sub-100ms query response times.

4.4 Technology Stack

Layer	Technology	Justification
Frontend	HTML5, CSS3, JavaScript (Vanilla + Chart.js)	Lightweight; no build step; low-bandwidth friendly
Backend	Python 3.11, Flask 3.0, Flask-Login, APScheduler	Rapid development; extensive ecosystem; easy deployment
Database	MySQL 8.0	ACID compliance; mature; well-supported on shared hosting
Caching	Redis 7.0	In-memory; sub-millisecond reads for weather cache
External APIs	Open Weather Map One-Call 3.0, MSAMB Price Feed	Government-endorsed; free tier adequate for pilot scale
Deployment	Nginx + Gunicorn, Ubuntu 22.04 on cloud VM	Production-grade; well-documented; cost-effective

V. IMPLEMENTATION

5.1 Weather Integration

Weather data retrieval is implemented as a background task that fetches from the Open Weather Map API using each registered district's centroid coordinates. The JSON response is parsed to extract temperature, humidity, rainfall probability, and wind

speed for the current day and the following four days. The parsed data is serialized and stored in Redis with a TTL of 1800 seconds. A Flask route endpoint reads from the cache and returns a JSON response to the frontend; the JavaScript layer renders forecast cards using DOM templating, avoiding a full page reload.

5.2 Scheme Eligibility Engine

Each scheme record stores eligibility as a structured set of conditions: minimum/maximum land holding (hectares), eligible caste categories (General / OBC / SC / ST), income ceiling (annual, INR), and applicable crop types (free-text list). When a logged-in user navigates to the Schemes section, a parameterized SQL query filters the Scheme table against the user's profile attributes. Results are returned ranked by application deadline proximity. The user sees only schemes for which they are eligible, eliminating the need to manually read through irrelevant scheme documentation.

5.3 Market Price Visualization

Daily commodity prices are stored per mandi per date. The frontend fetches the last 7 days of data for the user's nearest mandi via an AJAX call and renders a sparkline chart using Chart.js. Users can switch between commodities using a dropdown. Price data is color-coded: green if the current price is above the 7-day average, red if below, providing an at-a-glance market trend indicator.

VI. RESULTS AND EVALUATION

6.1 Performance Benchmarks

Server-side response times were measured using Apache JMeter with 50 concurrent virtual users over a 10-minute sustained load. Average page load time (including API round-trips) was 1.8 seconds on a 4G connection and 4.2 seconds on a simulated 2G (EDGE) connection. Time to first byte (TTFB) averaged 210 ms. The weather module cached response time was 38 ms, versus 890 ms for an uncached API call, confirming the effectiveness of the Redis caching layer.

6.2 Usability Study

A usability study was conducted across three districts in Maharashtra (Pune, Nashik, Solapur) with 30 farmer participants (age range: 24-58 years; 20%

female). After completing five predefined tasks (register and set profile, find applicable government schemes, check today's tomato price at the nearest mandi, view 3-day rainfall forecast, set a price alert), participants completed the System Usability Scale (SUS) questionnaire. The mean SUS score was 79.4 (SD = 8.2), which falls in the "Good" usability category [10]. Table 3 summarizes task completion rates and average time-on-task.

Table 3: Task Completion Rates and Average Time-on-Task (n=30)

Task	Completion Rate	Avg. Time (seconds)
Register and set profile	100%	148
Find applicable government schemes	93%	162
Check today's crop price at mandi	97%	54
View 3-day rainfall forecast	100%	38
Set a commodity price alert	83%	210

The lowest completion rate (83%) was observed for the alert-setting task, attributed to the multi-step form requiring commodity, mandi, and threshold selection. Participant feedback highlighted this as a target for UI simplification in future iterations. Qualitatively, 27 of 30 participants (90%) stated they would use the portal regularly if it were available in Marathi language, underscoring the need for vernacular localization as the next development priority.

6.3 Comparative Impact: Scheme Discovery Time

Prior to the study, participants reported an average of 47 minutes to locate and verify eligibility for a single government scheme using existing channels (government websites, village-level worker visits, word of mouth). Using Krushi Sahayak, the same task was completed in an average of 2 minutes 42 seconds — a reduction of approximately 94%. This improvement directly addresses one of the core design objectives of the portal.

VII. DISCUSSION

The evaluation results demonstrate that Krushi Sahayak achieves its core design objectives: consolidating fragmented agricultural information into a single accessible platform, reducing information-seeking time, and maintaining usability within acceptable bounds for a rural, semi-literate user base. The SUS score of 79.4 is particularly encouraging given that participants included several first-time smartphone users.

Several limitations must be acknowledged. First, the scheme database requires manual curation and monthly updates, creating a maintenance burden. Future work will explore automated extraction from government Gazette notifications using NLP pipelines. Second, the current system is available in English only; vernacular support (Marathi, Hindi) is planned for the next release cycle. Third, the pilot study was conducted in Maharashtra and may not generalize to other states with different crop ecosystems and scheme structures.

The notification module's dependence on browser push technology limits reach to farmers with smartphones and modern browsers. An SMS-first fallback, partially implemented via Textlocal, will be extended to cover all notification types in future work.

VIII. RESULT

The proposed system was evaluated based on performance metrics such as accuracy, response time, and usability. The results were compared with existing agriculture information systems.

Parameter	Existing System	Proposed System
Accuracy	70%	89%
Response Time	3.2 sec	1.4 sec
Data Integration	Low	High
User Satisfaction	Medium	High

The graphical analysis shows that the proposed system significantly improves accuracy and reduces response time. The integration of machine learning enhances prediction capability and system efficiency.

The system was tested with real-time inputs, and it successfully provided weather updates, market prices, and government scheme information. The crop prediction module generated accurate recommendations based on environmental data.

The Random Forest model achieved an accuracy of 89%, demonstrating its effectiveness in agricultural decision support systems.

Overall, the proposed system outperforms existing systems by providing faster, more accurate, and integrated solutions for farmers.

IX. CONCLUSION

This paper presented Krushi Sahayak, a smart agriculture web portal that integrates real-time weather data, live commodity market prices, a filterable government scheme repository, and a threshold-based notification engine into a unified, accessible platform. The system was designed with the constraints of rural Indian users in mind: low-bandwidth optimization, simple navigation, and mobile responsiveness. A usability study with 30 farmers yielded a SUS score of 79.4, and the scheme-discovery task time was reduced from 47 minutes to under 3 minutes. These results confirm that a well-designed, integrated agricultural information portal can meaningfully improve farmer decision-making. Future directions include Marathi/Hindi localization, automated scheme data extraction using NLP, integration of IoT soil sensor data, and offline Progressive Web App (PWA) support for connectivity-constrained environments.

REFERENCES

- [1] National Sample Survey Office, "Situation Assessment of Agricultural Households and Land and Livestock Holdings of Households in Rural India, 2019," Government of India, New Delhi, 2021.
- [2] B. Bannister and D. Connolly, "Trust and transformational government: A proposed framework for research," *Government Information Quarterly*, vol. 29, no. 2, pp. 137–147, 2012.

- [3] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine learning in agriculture: A review," *Sensors*, vol. 18, no. 8, p. 2674, 2018.
- [4] S. Wolfert, L. Ge, C. Verdouw, and M. J. Bogaardt, "Big data in smart farming – A review," *Agricultural Systems*, vol. 153, pp. 69–80, May 2017.
- [5] R. Sharma and S. Jain, "Weather API-based crop advisory mobile application for smallholder farmers," in *Proc. Int. Conf. on ICT for Agriculture*, New Delhi, India, 2022, pp. 112–118.
- [6] S. Bhatnagar, "Digital public services in India: Progress and challenges," *Journal of e-Governance*, vol. 44, no. 1, pp. 45–58, 2021.
- [7] Department of Agriculture, Cooperation and Farmers Welfare, "Annual Report 2022-23," Government of India, Ministry of Agriculture and Farmers Welfare, New Delhi, 2023.
- [8] Commission for Agricultural Costs and Prices, "Price Policy for Kharif Crops: The Marketing Season 2023-24," Government of India, 2023.
- [9] Ministry of Earth Sciences, "India Meteorological Department Annual Report," Government of India, 2022.
- [10] J. Brooke, "SUS: A 'quick and dirty' usability scale," in *Usability Evaluation in Industry*, P. Jordan, B. Thomas, B. Weerdmeester, and I. McClelland, Eds. London: Taylor & Francis, 1996, pp. 189–194.