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 7, October 2026
Impact Factor: 8.2

Smart Agricultural Marketplace: A Mobile-First Digital Platform for Direct Farmer-to-Buyer Trade with Real-Time Pricing and UPI-Based Payments

Aishwarya R. C.[1], Harsha Vardhan M. N.[2], Prof. Bharath Kumar B. S.[3], Prof. Divya C.[4]
[1,2]Students, Department of Information Science and Engineering
[3]Guide, Department of Information Science and Engineering
[4]Faculty, Department of Information Science and Engineering
Kalpataru Institute of Technology, Tiptur, Karnataka, India

Abstract: *Many farmers in India find it hard to sell their crops at a good price. They often depend on middlemen. They do not always know the prices in other markets. Payments can also come late. In this paper we describe the Smart Agricultural Marketplace. It is a mobile app that lets farmers and buyers deal with each other directly. Farmers list their crops. Buyers search the list and send requests or bids. The app also shows market prices and supports UPI payment. We plan to build the app with Flutter, and to use Firebase and Node.js for the backend. The app will work in more than one language, including Kannada and English. So far we have built a web prototype. It shows the seller and buyer screens: listing a crop, searching crops and tracking requests. Payment, live market prices and testing with real farmers are still to be done. We hope the app will help farmers find buyers, compare prices and sell with more confidence.*

Keywords: Agricultural marketplace, direct trade, crop listing, bidding, UPI payments, Flutter, Firebase, multilingual app.

I. INTRODUCTION

Agriculture is important for India. Many people depend on it for their income. But selling crops is not easy, especially for small farmers. They often have only a few buyers to choose from. They also do not always know what a fair price is. Most farmers sell through local traders and commission agents. Each person in the chain takes a share. So the price the farmer gets can be much lower than the price the consumer pays.

Information is another problem. A farmer who cannot compare prices in different markets finds it hard to decide on a good rate. Bargaining is also hard when there are few buyers. Payment can come late, and that hurts families who depend on farm income. Farmers also have few easy ways to talk to buyers directly or to find out which crops are in demand.

Today many villages have smartphones, mobile internet and UPI. This gives us a chance to make selling easier. An app can let a farmer show what is available for sale. Buyers can see it, compare offers and contact the farmer. UPI can be used for the payment.

In this paper we propose the Smart Agricultural Marketplace. It connects farmers with wholesalers, retailers and individual buyers in one app. A farmer lists a crop with its quantity, price, quality and location. A buyer searches the listings and sends an offer. Market prices help both sides while they talk. The app also supports regional languages for users who are not comfortable with English.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40435

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 7, October 2026
Impact Factor: 8.2

Our main work is the design of this marketplace. It brings crop listing, buyer requests, price information and direct communication into one app. We also built a small web prototype to show some of the screens. The mobile app, payment, live price data and field testing are future work.

The rest of the paper is arranged as follows. Section II looks at existing platforms and related research. Section III explains the problem and Section IV gives our aim and objectives. Section V describes the proposed system. Section VI covers the architecture and the development steps. Section VII discusses implementation and shows the prototype. Section VIII gives our testing and deployment plan. Section IX concludes and lists future work.

II. RELATED WORK

A. Existing Digital Agriculture Platforms

Some digital services already work in Indian agriculture. They do different jobs. Some give advice, some supply farm inputs, some handle buying and delivery, and some sell to consumers.

The National Agriculture Market (eNAM) is an online trading system of the Government of India. It links agricultural markets and helps share price information. It works together with the existing markets. It does not replace them [2].

AgroStar gives farming advice and sells farm inputs [3]. DeHaat offers advice, inputs and links to markets [4]. Ninjacart buys produce and supplies it to business customers [5]. BigBasket is an online grocery store. It gets its produce through its own buying network [6].

All these services use technology, but for different purposes. Our marketplace is different in one way. The farmer makes the listing, buyers send offers, and both sides can talk directly. Price information and a multilingual screen are in the same app. How well it works will depend on how well we build it. It will also depend on how many people use it, how correct the price data is, and whether payment and delivery can be arranged.

B. Related Academic Research

Many researchers have used technology in farming. They have used machine learning, sensors and data to help with farm decisions. Examples are soil analysis, crop monitoring and choosing which crop to grow.

One example is Soil-MobiNet. Gyasi and Purushotham built a convolutional neural network that classifies soil from photos taken with a smartphone. It also records where the photo was taken [1].

This kind of work is useful background. But it is about the farm, not about selling. Our app is about selling. So it starts with listings, buyer search, offers and managing deals. Later we may use machine learning to estimate crop prices or spot price trends. For that we would need good past price data. We would also need to check the results on real market data.

III. PROBLEM STATEMENT

Selling farm produce has several problems for farmers and buyers. Our project tries to solve these:

- **Few buyers.** Farmers often depend on local traders. It is hard for them to find other buyers by themselves.
- **Not enough price information.** Farmers cannot easily see prices from many markets. So it is hard to judge an offer or to bargain well.
- **Weak bargaining power.** Small farmers have little say when only a few buyers or middlemen are available.
- **Payment problems.** Late payment makes it hard to pay household costs and to prepare for the next crop.
- **No direct contact.** Farmers and buyers cannot easily talk about quality, quantity, availability and delivery.
- **Hard to use digital services.** Language, little experience with apps and weak internet stop some rural users from using online platforms.

These problems are why we want to build a digital marketplace. It should help people find crops, talk directly, compare offers and manage deals. We try to do this with a simple screen and a modular design.

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 7, October 2026
Impact Factor: 8.2

IV. AIM AND OBJECTIVES

A. Aim of the Project

The aim is to design a digital marketplace where farmers and buyers can find each other and agree on a price for farm produce. We want to give better access to market information, make price talks fair and open, and make communication between the two sides easier.

B. Objectives

- Give farmers a way to publish listings of their produce.
- Let buyers search the available crops and send purchase offers.
- Show market-price information to help both sides negotiate.
- Add a bidding feature so a farmer can compare offers.
- Support digital payment through a UPI payment provider.
- Build a multilingual interface for users with different language needs.
- Allow direct communication between farmers and interested buyers.
- Design the system so that new services can be added in later versions.

We will know whether these objectives are met only after we build the full system and test it with users.

V. PROPOSED SYSTEM

The Smart Agricultural Marketplace is a platform where farmers can show what they have to sell and buyers can find what they need. It should also save people from visiting many markets just to find out what is available or to compare first offers. The main features are described below.

A. Crop Listing

A farmer can create a listing with the crop name, quantity, expected price, quality, harvest date and location. This helps a buyer decide if the crop suits them. The farmer should be able to change the listing when the stock changes or when there is a mistake. Clear details also avoid confusion later.

B. Buyer Search and Bidding

Registered buyers can look through the listings and search by crop or location. A buyer can send an offer for a crop. The farmer sees the offers and picks one. The choice depends on price, quantity and other needs. A fixed bidding period and a clear status for each offer will keep the process in order.

C. Market-Price Information

The app will show market prices. A farmer or a buyer can compare an asking price or an offer with them. This feature is only as good as its data. So each price should show where it came from, which market it is for and the date. We will call the prices real-time only if the app gets fresh data often enough.

D. Multilingual Interface

The app will work in more than one language so that people can use it in the language they know. Labels, instructions, messages and statuses should be translated in the same way everywhere. Simple menus, easy text and clear icons will help people who are new to smartphones.

E. Digital Payment Support

We plan to use UPI for payment between the buyer and the seller. A payment provider such as Razorpay can start and check the payment. The app will save the payment reference and status. It will tell both people when the provider confirms the payment. Payment is a planned feature. It is not built yet.

F. Direct Communication

Chat will let a farmer and a buyer talk about quantity, quality, availability and delivery before they finish a deal. It can also help them work together again later. Only the two people in a deal should be able to see their chat and deal details. Together these features make a marketplace where it is easier to find produce and where trade has a clear structure.

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 7, October 2026
Impact Factor: 8.2

VI. SYSTEM ARCHITECTURE AND METHODOLOGY

We follow these steps: requirement analysis, design, development, testing and a small pilot. Table I lists them.

Table I: Development stages of the Smart Agricultural Marketplace.

Phase	What we do
Requirement Analysis	Find out what farmers and buyers need. This includes language, internet limits, ease of use and safe payments.
System Design	Plan the app structure, the user steps, the database and how the modules work together.
UI/UX Design	Make simple screens with clear labels, easy menus and more than one language.
Development	Build the mobile app in Flutter and the backend with cloud and server tools.
Database Setup	Store user profiles, crop listings, buyer offers, requests and payment records.
Payment Integration	Connect a UPI payment provider. Decide how to confirm payments and handle failures.
Testing	Check that the app works, is easy to use, is reliable and is secure, also on weak networks.
Deployment	Give the app to a small group of users, take their feedback and improve it before a wider release.

A. High-Level Architecture

The system has a client side and a server side. The user screen gives access to the marketplace. The backend stores data, runs the logic and talks to outside services.

We plan to use Flutter for the mobile app. One codebase can run on both Android and iOS. This makes the app easier to maintain.

Firebase can handle login, the database, live updates and image storage. A Node.js backend can handle extra logic, such as checking offers, handling payments and talking to other services.

The app and the backend will talk over HTTPS. Login and permission checks will protect user accounts and records.

The design is modular, so each part can be built and maintained on its own. The exact choice of Firebase database, Node.js endpoints and payment interface will be settled during development.

B. Core Functional Modules

- **User and Role Management:** Handles sign-up, login and access for farmers and buyers. Each role sees its own dashboard.
- **Crop Listing Module:** Lets farmers create, update and manage listings with crop name, quantity, price, quality, harvest date and location.
- **Bidding Module:** Lets buyers send offers on listings. The system checks each offer and saves its status. The farmer reviews the valid bids and accepts one.
- **Market-Price Module:** Shows market prices from known sources. It keeps the crop, market, unit and date for each price so users can read it correctly.
- **Payment Module:** Sends payment requests and saves the result. It keeps initiated, successful, failed and pending payments apart. It does not assume a request is paid.
- **Messaging Module:** Gives farmers and buyers a way to chat. Only the right users can see the messages.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40435

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 7, October 2026
Impact Factor: 8.2

- **Localisation Module:** Manages translated text and language choice. The app remembers the chosen language when the user comes back.

VII. IMPLEMENTATION CONSIDERATIONS

A. Technology Stack

We plan to use Flutter for the app screens. The same code can run on different mobile systems. Whether it suits our project will depend on the screens we need, the phones we support and our test results.

Firebase can give login, a database and file storage. Node.js can run the server tasks that need more logic. We will choose the database setup after we know our data, our queries and how much live updating we need.

For payment we may use a UPI provider such as Razorpay. Before we release the app, we must check its integration needs, fees, payment flows and security rules.

These technologies are our planned stack. This does not mean all of them are already in the prototype.

B. Payment Security and Reliability

Payment needs special care. A mistake can cause loss of money or confusion between the buyer and the seller. We will use a well-known payment provider and check the payment status on the server. A success message on the phone alone is not proof that the money was paid.

Each payment will have its own reference and a clear status. The app must handle broken connections, repeated requests, failed payments and late confirmations. It must not record the same payment twice. Login, role-based access, safe communication and careful handling of records also matter. Before release we will test successful, failed and interrupted payments.

C. Multilingual and Low-Connectivity Design

Rural users differ in language, in how well they know smartphones and in how good their internet is. So we want the menus and forms to be simple. Translated labels, clear instructions, easy text and the same icons everywhere will help. Voice input could be added later.

The app should also work when the network is weak. It can save recently viewed data on the phone so users can read it offline. Actions that change data on the server, such as sending a bid or a payment request, need care when the connection breaks. We will retry them only in a controlled way, with checks to avoid duplicates. We think these steps will help, but we still have to confirm it by testing with real users and real networks.

D. Prototype Implementation Results

We built a small web prototype called sm.html to show some marketplace screens in a desktop browser. It has separate entry points for buyers and sellers, and each one opens its own dashboard. The prototype shows basic listing and product search. It is not the full mobile app described above. Payment, live market prices and tested multilingual support are not part of it. Figs. 1 to 3 show the screens.

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 7, October 2026
Impact Factor: 8.2

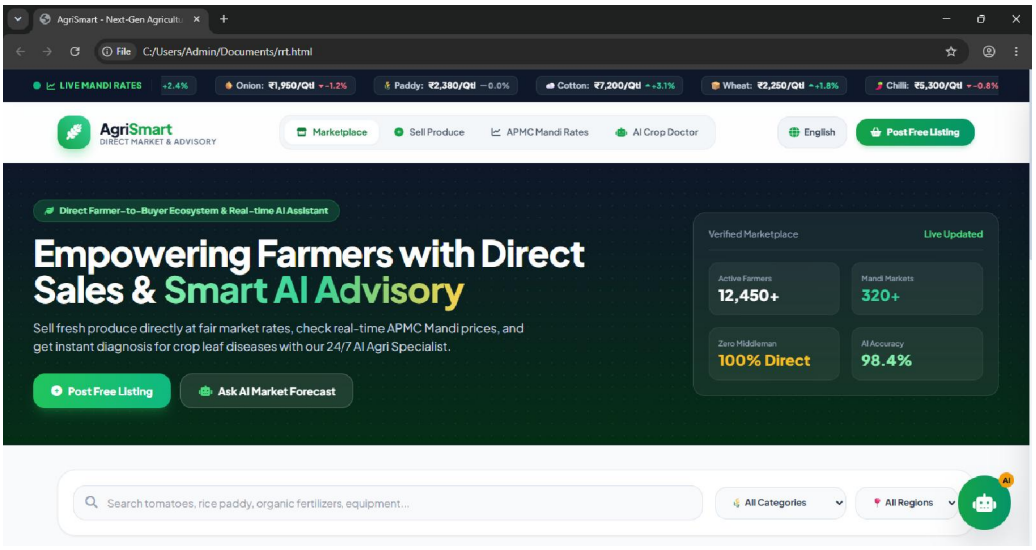


Fig. 1: Landing page of the Smart Agricultural Marketplace.

The landing page lets the user continue as a buyer or as a seller. The choice opens the matching dashboard. This is the basic role-based navigation of the prototype.

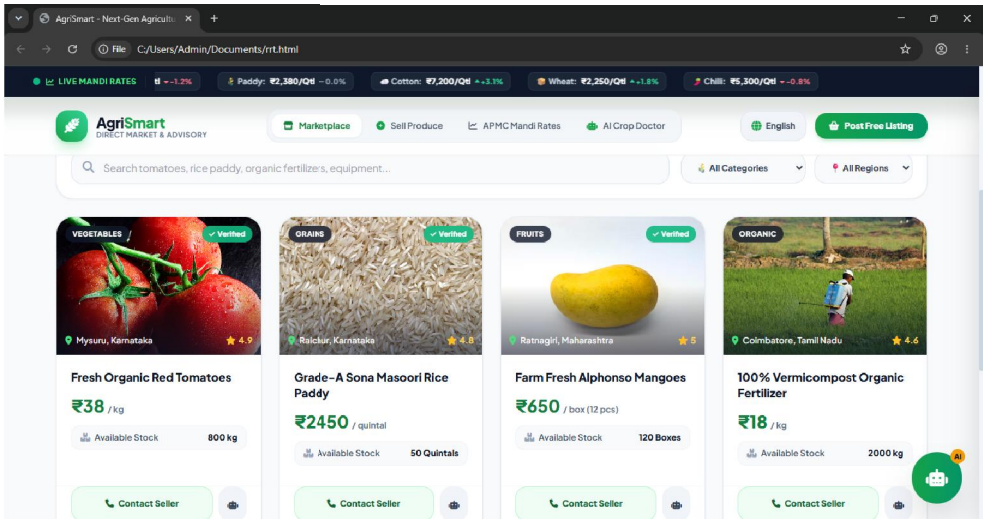


Fig. 2: buyyer marketplace

The Seller Dashboard (Fig. 2) shows the farmer's crops in a table. It lists crop name, quantity, harvest date, price per kg, total price and location. A form below lets the seller add a new crop. The total price is the quantity multiplied by the price per kg. For example, 467 kg of maize at ₹15.50 per kg gives ₹7,238.50, as shown in the table. This only shows the listing screen. It is not a completed sale.

Copyright to IJARSCT
www.ijarsct.co.in
DOI: 10.48175/IJARSCT-40435

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 7, October 2026
Impact Factor: 8.2

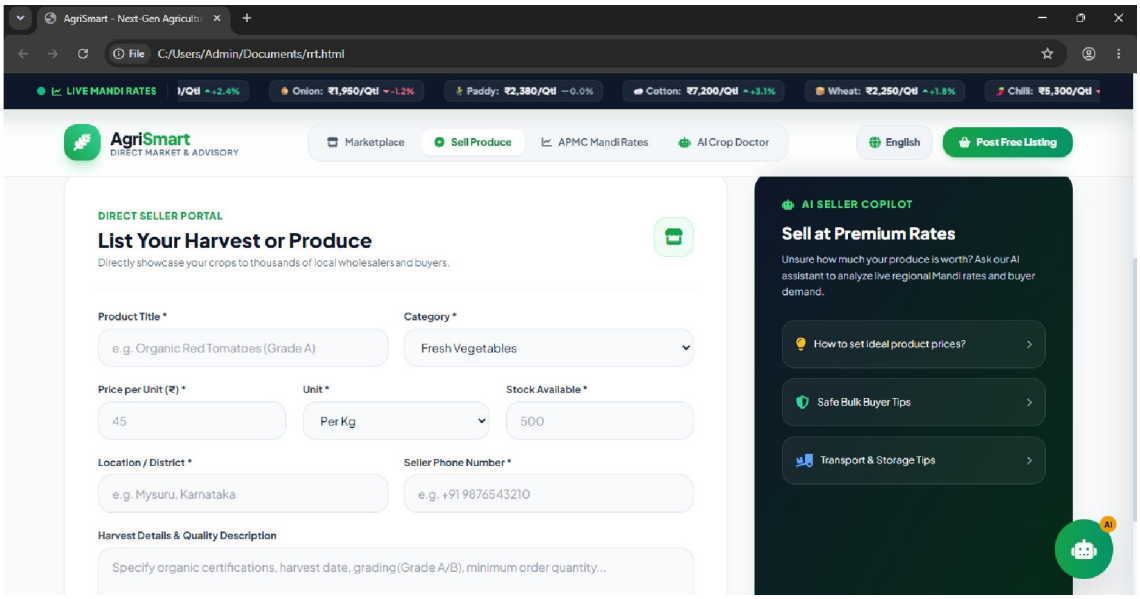


Fig. 3: seller Dashboard.

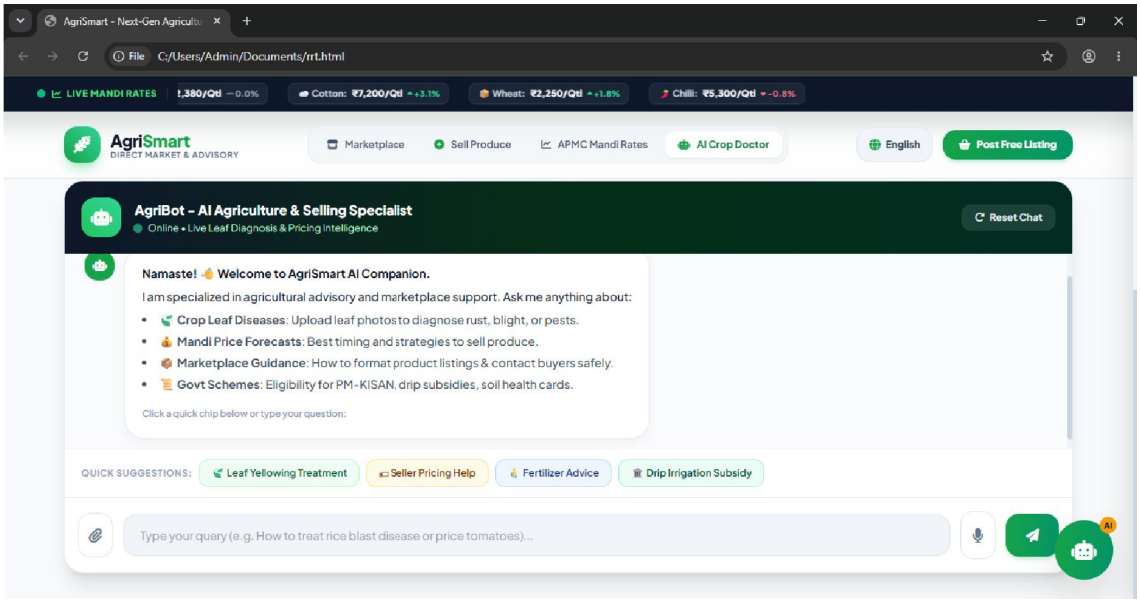


Fig 4 :AI agricultural selling assistant

The Buyer Dashboard (Fig. 3) shows the requests the buyer has sent and their status, such as pending. It has a search box and a list of crops for sale. A buyer can use these to find products and check details before sending a request. The prototype shows the screen and the basic flow only. A real bidding and payment system still has to be built and tested.

In short, the prototype shows how seller and buyer activities are kept apart. We still need to connect the full backend, check the transaction flows and test the system in real conditions.

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 7, October 2026
Impact Factor: 8.2

VIII. TESTING AND DEPLOYMENT APPROACH

We need a proper testing plan to check that the app meets its needs. We plan the following tests.

- **Functional testing.** Check that users reach the right dashboard, create listings, search crops, send requests and see updated status. Test wrong values and incomplete forms too.
- **Bidding workflow testing.** Check offer validation, bid submission, status updates and how an offer is accepted. The system should block invalid or duplicate actions.
- **Payment testing.** Test successful and failed payments, late confirmations, broken connections and repeated requests. A payment is recorded only after it is verified.
- **Usability testing.** Ask real farmers and buyers to do common tasks in the app. Their feedback will show problems with menus, words, language choice and data entry.
- **Connectivity and reliability testing.** See how the app behaves on slow networks, short disconnections and interrupted requests. It should show clear messages and never lose a user's action silently.
- **Security testing.** Check login, role permissions, data checks and protection of user and payment records.

After internal testing, we could run a small pilot with a few farmers and buyers in one area. Their feedback will help us find technical problems and improve the screens before a wider release.

This paper gives the plan only. We have not done field testing yet. We will add test results and user feedback after the tests are done.

IX. CONCLUSION AND FUTURE WORK

This paper described the design of the Smart Agricultural Marketplace. It is meant to help farmers and buyers deal with each other directly through an app. The app combines crop listing, product search, offers, market prices, chat and UPI payment in a multilingual interface.

It tries to solve some real problems in selling farm produce: few buyers, little price information, weak bargaining and trouble managing deals. By giving one place to find crops and talk to buyers, it aims to make trade easier and more open.

We built a web prototype that shows some seller and buyer screens: crop listing, price calculation, product search and request tracking. The full mobile app, payment, checked price data and field testing still need to be done. So the benefits we mention are what we expect. They are not proven results.

We have some ideas for later. Machine learning could estimate crop prices from past market data. Links with transport services could help move the crop after a deal is agreed. Warehouse and cold-storage services could help with unsold produce. Weather information, government scheme updates and farming tips could also be added.

Next we will focus on safe payments, reliable price data, easy regional-language screens and testing with real users. This will show whether the app is useful to farmers and buyers in real markets.

REFERENCES

[1] E. K. Gyasi and S. Purushotham, "Soil-MobiNet: A convolutional neural network model base soil classification to determine soil morphology and its geospatial location," Sensors, vol. 23, no. 15, Art. no. 6709, 2023, doi: 10.3390/s23156709.

[2] Government of India, "National Agriculture Market (eNAM)," Ministry of Agriculture and Farmers Welfare. [Online]. Available: https://www.enam.gov.in

[3] AgroStar, corporate platform information. [Online]. Available: https://www.agrostar.in

[4] DeHaat, corporate platform information. [Online]. Available: https://www.dehaat.com

[5] Ninjacart, corporate platform information. [Online]. Available: https://www.ninjacart.com

[6] BigBasket, corporate platform information. [Online]. Available: https://www.bigbasket.com