



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 10, October 2026

AgroSmart: A Unified Web Platform for Direct Crop Trading, Farm Input Procurement, Farm Financial Management and AI-Assisted Crop Diagnosis

Dinesh Deore, Sushant Ashok Pandey, Shaikh Mohd Kaif Asfaque Ahmed, Saizan Salimkadir Mulla,
Sameer Choudhary

Department of Computer Engineering, Rizvi College of Engineering, Mumbai, Maharashtra, India

ABSTRACT: Farmers commonly depend on several disconnected channels to sell their harvest, to buy seeds, fertilizers and equipment, and to keep records of income and expenses. This fragmentation limits market visibility, makes price comparison and negotiation difficult, delays order coordination and leaves farm finances poorly documented. Studies of digital agricultural markets show that their benefit depends on trust, buyer relationships and market information, and existing platforms rarely combine crop selling, input procurement and financial record keeping. This paper presents AgroSmart, a unified role-based web platform that connects farmers directly with wholesalers, retailers, industries and consumers, and that also provides a store for certified farm supplies from verified sellers. Farmers list crops with images, quantity, grade, expected price, harvest date and location; buyers search, compare and negotiate prices through offers; and orders, payments and delivery status are tracked end to end. A financial dashboard turns sales, expenses, purchases and yield records into profit and loss indicators, interactive charts and downloadable reports, and an AI Crop Doctor module gives image-based diagnosis of crop leaf problems with treatment guidance and Hindi and English audio readout. The platform uses a React and TypeScript client, a FastAPI REST backend and a PostgreSQL database, with token-based authentication, role-based access control and atomic order processing. The system has been fully implemented, and its main modules are presented through architecture diagrams and screenshots of the working application. The paper focuses on the design and implementation of the platform and does not report quantitative experimental results.

KEYWORDS: Smart Agriculture, Agricultural E-Commerce, Digital Marketplace, Price Negotiation, Farm Financial Management, Crop Disease Diagnosis, Role-Based Access Control, FastAPI, PostgreSQL.

I. INTRODUCTION

Agriculture is the primary livelihood for a large part of the population in India, yet many farmers still sell their crops through local markets and intermediaries and have limited visibility of the prices offered elsewhere. Purchasing inputs such as seeds, fertilizers, crop-protection products and tools involves separate local sellers whose price, quality and availability vary. Income and expenses are often noted informally, so a farmer has difficulty judging the profit or loss of a season, comparing the cost of inputs or planning the next purchase.

Online marketplaces and digital payments have changed trade in many sectors through searchable catalogues, structured orders and traceable transactions. Agriculture, however, has requirements that a general-purpose marketplace does not address well. Listings depend on harvest date, quantity, grade and location; prices are frequently negotiated rather than fixed; a farmer is at the same time a seller of crops and a buyer of inputs; and financial planning is tied to seasons and inputs. In addition, many farmers need practical advice when crops show signs of disease, and the advice is most useful when it is available in their own language.

Prior work indicates that digital agricultural markets can benefit farmers, but that the benefit depends on factors such as trust, buyer relationships and the quality of market information [1], [2]. These findings suggest that a platform should

make prices comparable, let parties negotiate in a traceable way, keep communication and order status visible, and give the farmer tools for managing the business side of farming.

To meet these needs, this project introduces AgroSmart, a unified web platform that brings together (i) a direct crop marketplace with search, comparison and price offers, (ii) an agricultural supplies store with cart and checkout, (iii) order and delivery management with notifications and messaging, (iv) a farmer financial dashboard, and (v) an AI Crop Doctor for image-based crop diagnosis. The main contributions of this paper are the integrated role-based architecture, the offer and order workflows expressed as explicit state machines, the transactional order-processing design that keeps payments and stock consistent, and a description of the completed system supported by screenshots of its working modules.

II. LITERATURE REVIEW

Argade et al. studied electronic agricultural markets in India using a survey-based exploratory factor analysis, comparing transaction costs in physical and electronic markets and considering the role of relational marketing [1]. The study provides empirical evidence from the Indian context and indicates that adopting an electronic market does not by itself guarantee lower transaction costs when market information and buyer relationships remain uneven. This motivates transparent listings, recorded offers and buyer-seller messaging in AgroSmart.

Do Prado et al. presented a qualitative case study of a digital marketplace that connects Brazilian small farmers with end consumers and analysed the factors behind its success [2]. The study shows that direct digital channels are practical for small producers and emphasises non-technical factors such as trust, stakeholder needs and management. It does not prescribe a technical architecture and does not combine crop selling with input purchasing or financial analytics.

Research on image-based assessment of produce is a second relevant strand. Akter et al. reviewed nondestructive sensing technologies, including imaging and machine-learning approaches, for measuring the external quality of fruits and vegetables [3]. Vignati et al. reviewed the concepts and applications of hyperspectral imaging for the quality assessment of fresh-cut produce [4]. These works support the feasibility of image-based assessment, while noting that outcomes depend on image conditions, validation and clearly defined quality measures, and that hyperspectral imaging typically requires specialised equipment that is less accessible for ordinary farm use than standard camera images.

On the technical side, a marketplace that handles money and stock needs reliable transaction handling. The ACID properties of relational databases, formulated by Haerder and Reuter in their work on transaction-oriented recovery, ensure that a group of related updates is either fully applied or not applied at all [5]. Access to different functions for different types of users is commonly governed by role-based access control, a model surveyed by Sandhu et al. in which permissions are attached to roles rather than to individual users [6]. Services are exposed through the REST architectural style described by Fielding [7], and signed JSON Web Tokens defined in RFC 7519 allow stateless authentication between client and server [8].

In summary, the literature supports the usefulness of digital agricultural markets and provides established building blocks for secure and consistent platforms, but existing studies generally treat crop marketplaces, input procurement, financial management and image-based advisory separately. AgroSmart combines them in a single role-based platform.

III. PROBLEM STATEMENT

Farmers need a single, convenient platform to sell harvested crops, purchase agricultural inputs and manage their transactions. In fragmented systems, farmers and buyers face limited market visibility, difficulty comparing prices, informal and untraceable negotiation, and delays in order coordination.

Selling through local channels exposes farmers to limited price information, while buyers such as wholesalers, retailers and industries cannot easily search for produce by crop, quantity, grade, harvest date and location. Negotiation usually happens by phone, so agreed prices and quantities are not recorded in a form that can be tracked. Buying inputs adds

another set of separate sellers with no common catalogue or order record, and the resulting purchases and expenses are rarely compared with the income of the same season.

Furthermore, when crops show signs of disease, farmers often lack quick access to reliable guidance on symptoms, treatment and prevention, and written advice in a foreign language is of limited use in the field.

Therefore, a need exists for an end-to-end platform that provides searchable and comparable listings, a recorded negotiation process, a store with cart and checkout for inputs, reliable order, payment and delivery handling, a financial view of the farm, and simple image-based crop guidance in local languages. The platform must also separate what each type of user may do, and it must keep stock and payment records consistent when several users act at the same time.

IV. GOALS & OBJECTIVES

Goal 1: Build a Direct Crop Marketplace

- Objective 1: Allow farmers to create crop listings with images, quantity, grade, expected price, harvest date and location.
- Objective 2: Provide search, filters and comparison for buyers such as wholesalers, retailers, industries and consumers.

Goal 2: Provide Structured Price Negotiation and Order Management

- Objective 1: Let buyers submit offers on listings and let farmers accept, reject or counter them, with unanswered offers expiring.
- Objective 2: Convert accepted offers into orders and track each order through confirmation, dispatch and delivery, with notifications and buyer-seller messaging.

Goal 3: Support Farm Input Procurement

- Objective 1: Let verified sellers publish seeds, fertilizers, crop-protection products, tools and equipment with price and stock.
- Objective 2: Provide cart, checkout and sandbox payment handling with consistent inventory updates.

Goal 4: Deliver Farm Financial Management

- Objective 1: Record crop sales, input purchases and manual expenses, and capture crop yield per acre.
- Objective 2: Present profit and loss, monthly revenue and expense charts, expense categories and productivity, with CSV export and print options.

Goal 5: Provide AI-Assisted Crop Diagnosis and Secure Access

- Objective 1: Offer an AI Crop Doctor that returns a diagnosis with confidence and severity, symptoms, organic and chemical control suggestions, preventive practice and audio readout in Hindi and English.
- Objective 2: Enforce token-based authentication and role-based access control for the farmer, buyer, seller and administrator roles.

V. APPROACH

1. Needs Assessment:

- Identify the user roles: farmer, buyer (wholesaler, retailer, industry or consumer), agricultural seller and administrator (Fig. 2).
- Analyse the problems of fragmented selling, informal negotiation, separate input procurement, informal record keeping and limited access to crop guidance.
- Define functional requirements for each role and non-functional requirements such as security, consistency, responsiveness and ease of use on phones and desktops.

2. Design and Development:

- Adopt a layered client-server architecture: a React and TypeScript web client styled with Tailwind CSS and using Recharts for charts, a FastAPI backend exposing REST services, and a PostgreSQL database accessed through SQLAlchemy (Fig. 1).
- Organise the backend into modules for authentication and role-based access control, crop listings and offers, supplies store with cart and orders, financial analytics, messaging and notifications, and the AI Crop Doctor.
- Model the main entities: users, crop listings, products, offers, carts, orders, order items, payments, messages, notifications, expenses, yield records and transactions (Fig. 4).

AgroSmart System Architecture

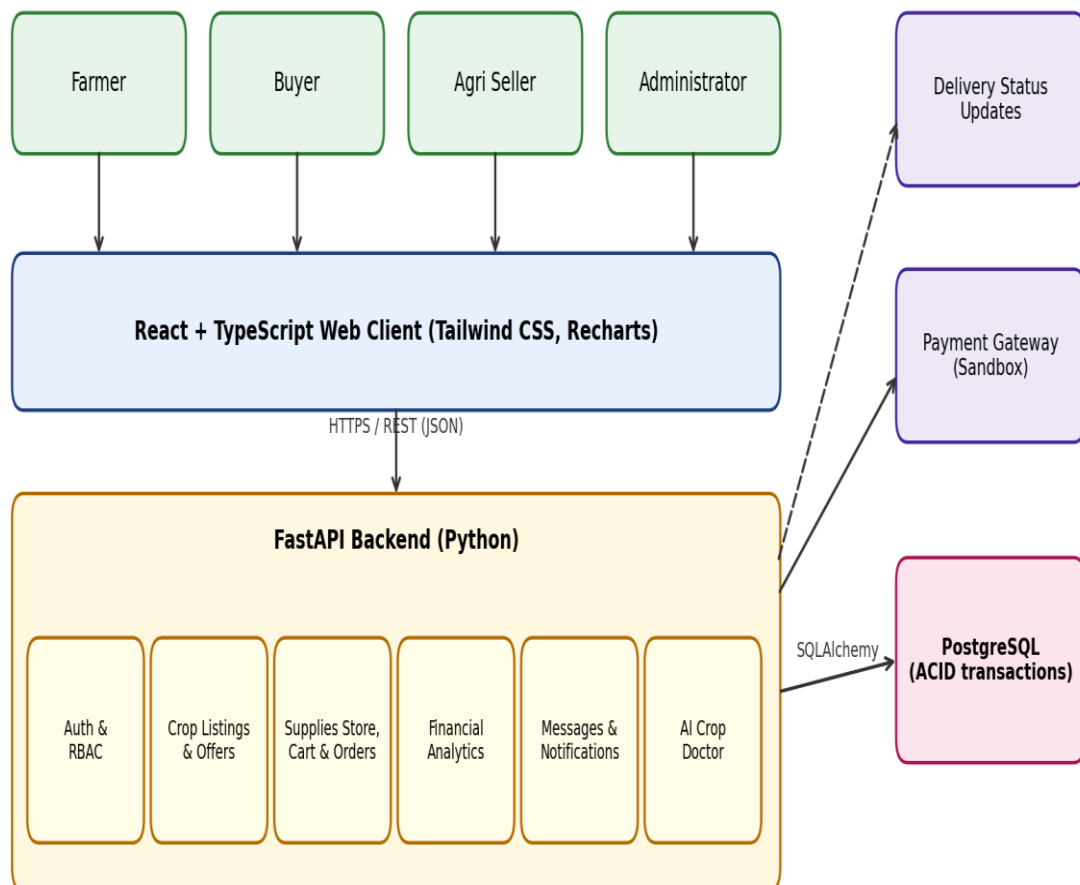


Figure 1: AgroSmart System Architecture and Module Topology

The diagram shows the four user roles accessing the React client, which communicates with the FastAPI backend over HTTPS. The backend contains the authentication, marketplace, supplies store, financial analytics, messaging and AI Crop Doctor modules, and persists data in PostgreSQL. A payment gateway in sandbox mode and delivery status updates complete the order flow.

Use Case Diagram of AgroSmart

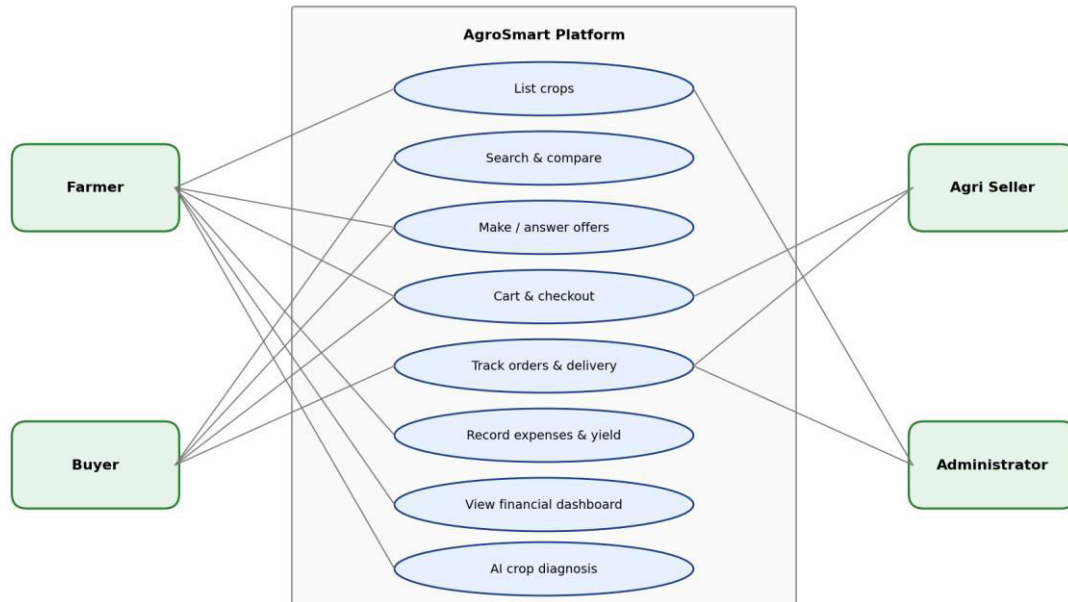


Figure 2: Use Case Diagram of the Four User Roles

The diagram summarises the main functions available to the farmer, buyer, agricultural seller and administrator roles, including listing crops, searching and comparing, making and answering offers, cart and checkout, order tracking, expense and yield recording, the financial dashboard and AI crop diagnosis.

3. System Implementation:

- Authentication and role-based access: the backend verifies the password against its stored hash and issues a signed token that carries the user identifier and role; every protected request verifies the token and compares the role with the permission of the endpoint. The interface also provides a demonstration mode in which the role can be switched between farmer, buyer, seller and administrator.
- Crop marketplace: listings are validated for required fields, positive quantity and price, valid harvest date and acceptable image type and size. Buyers search by crop and filter by location, price range, quantity and harvest date, and listing cards show the grade, price per unit, available quantity and minimum order.
- Offer negotiation: a buyer submits an offer, and the farmer accepts, rejects or counters it; unanswered offers expire, and an accepted offer creates an order at the agreed price (Fig. 3).
- Supplies store and orders: verified sellers publish farm inputs; buyers and farmers use a cart and checkout, and each order carries its payment status and delivery status.

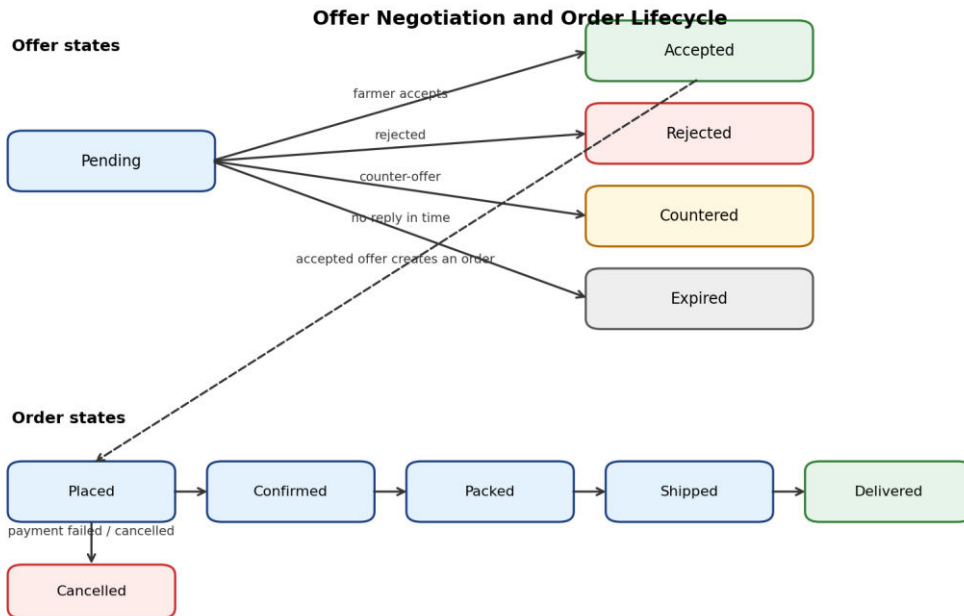


Figure 3: Offer Negotiation and Order Lifecycle State Diagram

An offer can be pending, accepted, rejected, countered or expired. An order moves through placed, confirmed, packed, shipped and delivered, and it is cancelled when a payment fails or the order is cancelled.

Entity-Relationship Overview (Simplified)

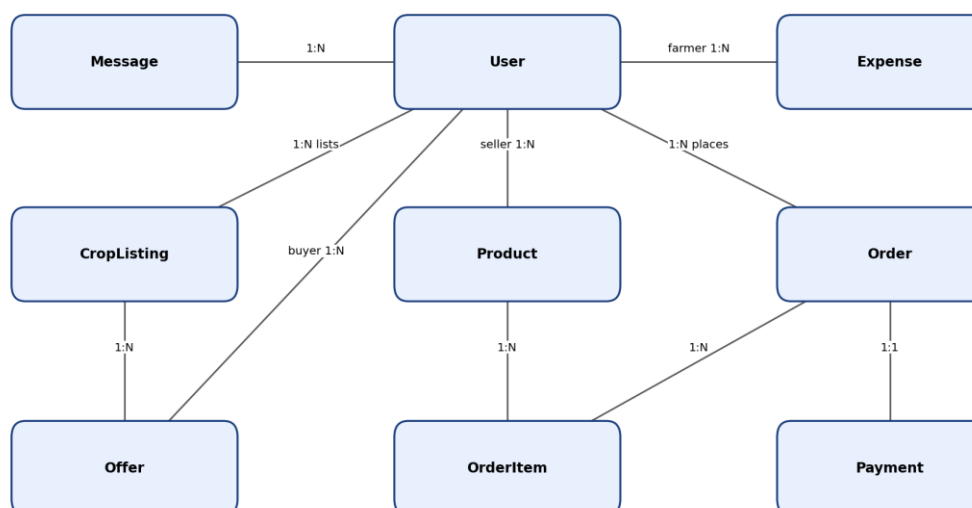


Figure 4: Simplified Entity-Relationship Overview

A user can own listings, products, offers, orders, messages and expenses; an order contains order items and has a payment; and order items refer to products.

4. System Optimization:

- Transactional consistency: at checkout the backend verifies stock for every item and creates the order and a payment record inside one database transaction. After a successful payment status in sandbox mode the order is confirmed and stock is reduced; a failed payment or a cancellation rolls back the transaction or restores stock.
- Concurrency control: row-level locking on product rows prevents two simultaneous checkouts from selling the same remaining stock.
- Query efficiency: results are paginated and sorted by relevance, price or date, and indexes are placed on frequently filtered columns.
- Validation and security: passwords are stored only as hashes, all input is validated on the server, sellers are verified before listing products, and card details are handled by the payment gateway rather than stored by the application.

5. Integration of Financial Analytics and AI Crop Diagnosis:

- Financial dashboard: completed crop sales, purchases from the supplies store and manually recorded expenses are stored as transactions. For a selected period and farm, the module reports gross crop sales, total farming expenses, farm supplies purchased and net profit or loss, together with the margin. Net profit or loss is revenue minus expenses, the margin is $(\text{revenue} - \text{expenses}) / \text{revenue} \times 100$ when revenue is greater than zero, and productivity is the total harvest divided by the cultivated area. Charts show monthly revenue and expenses and the expense categories, and the records can be exported as CSV or printed (Fig. 5).
- AI Crop Doctor: the farmer selects a crop and uploads a field photograph or chooses a sample photograph. The module analyses the image with computer-vision pattern matching against an agronomy knowledge base and returns the diagnosis with a confidence score and severity, the observable symptoms, organic and biological control, approved chemical dosage, preventive practice and an audio readout in Hindi or English (Fig. 6). The result is advisory, and the interface advises confirming it with a local agricultural specialist.

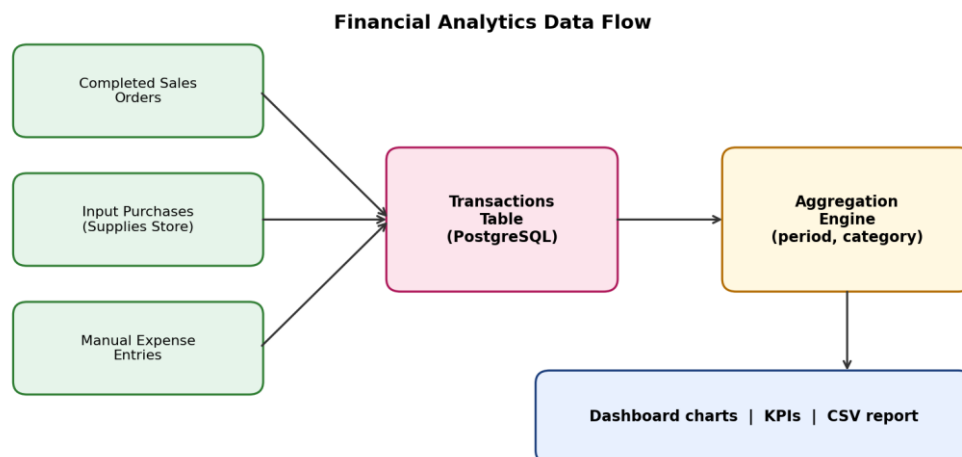


Figure 5: Financial Analytics Data Flow

Completed sales orders, input purchases from the supplies store and manual expense entries flow into the transactions table. The aggregation service summarises them by period and category and feeds the dashboard charts, the key indicators and the CSV report.

AI Crop Doctor: Diagnosis Flow

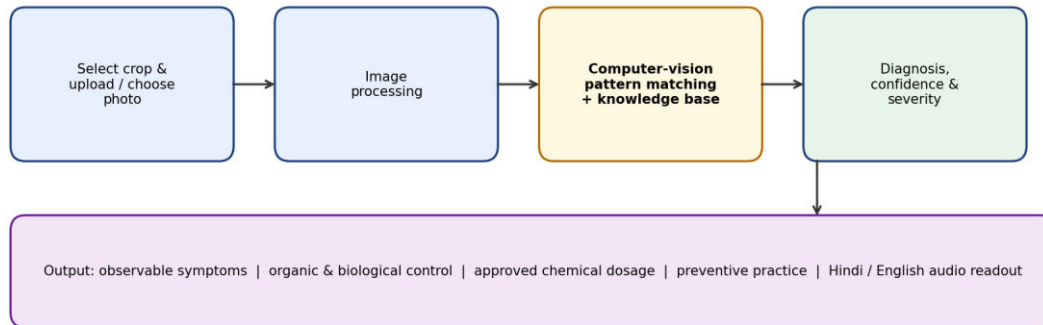


Figure 6: AI Crop Doctor Diagnosis Flow

The farmer selects the crop and supplies a photograph; the image is processed and matched by the diagnosis module, which returns the diagnosis with confidence and severity along with symptoms, control measures, preventive practice and audio readout.

6. Testing and Evaluation:

- Unit tests of business logic such as validation, offer states and the financial formulas, and API tests of the main endpoints.
- Integration tests of the offer, checkout and payment flow, and concurrency tests in which several checkouts compete for the same stock.
- Role-based tests that confirm each role can reach only its permitted functions.
- Usability checks of the main screens on phone and desktop browsers.
- The measures used are the share of test cases passed, API response time for search queries, task completion and user satisfaction. Quantitative results are not reported in this paper.

Code Snippets: the following listings are simplified illustrations of the logic of the checkout transaction and of the financial summary.

Checkout with stock check and rollback (Python/FastAPI, simplified illustration):

```

@router.post("/orders/checkout")
def checkout(cart_id: int, db: Session = Depends(get_db),
             user: User = Depends(require_role("buyer", "farmer"))):
    try:
        items = get_cart_items(db, cart_id, user.id)
        for it in items:
            product = db.get(Product, it.product_id, with_for_update=True)
            if product.stock < it.quantity:
                raise HTTPException(status_code=409, detail="Insufficient stock")
        order = create_order(db, user.id, items)
        create_payment(db, order.id, status="initiated")
        db.commit()
    except Exception:
        db.rollback()
        raise
    return {"order_id": order.id, "status": order.status}
  
```

Listing 1: Checkout Endpoint Creating the Order and Payment in One Transaction

Financial summary (Python, simplified illustration):

```
def finance_summary(sales, expenses, area_acres, quantity):
    revenue = sum(s.price * s.quantity for s in sales)
    total_expense = sum(e.amount for e in expenses)
    profit = revenue - total_expense
    margin = (profit / revenue * 100) if revenue > 0 else 0.0
    productivity = (quantity / area_acres) if area_acres > 0 else 0.0
    return {"revenue": revenue, "expenses": total_expense,
            "profit": profit, "margin_percent": margin,
            "productivity": productivity}
```

Listing 2: Calculation of Revenue, Expenses, Profit, Margin and Productivity

VI. FUTURE SCOPE

The AgroSmart platform provides a complete foundation for direct agricultural trading and farm management. In subsequent development cycles, several advancements are planned to expand its functionality:

1. **Production payments and logistics:** replacing the payment sandbox with a production gateway and integrating a logistics provider for real-time delivery tracking.
2. **Wider language and low-literacy support:** extending the Hindi and English interface and audio support to more regional languages and simpler voice-driven navigation.
3. **Expanded AI models:** adding more crops and conditions to the crop diagnosis module, training with larger and more varied field images, and reporting measured accuracy.
4. **Price analytics:** price-trend views and forecasts for crops once enough reliable transaction data has been collected.
5. **Seller verification and dispute handling:** stronger verification steps, ratings, audit logs and a dispute-resolution workflow.
6. **Mobile application:** a companion mobile application with push notifications and offline entry of expenses and yield for low-connectivity areas.

Proposed System and Implementation Outputs

The AgroSmart platform has been fully realised as a web application. Below are the screen captures of the working system. The screens were captured in demonstration mode, in which the user role can be switched between farmer, buyer, seller and administrator, and the data shown is demonstration data.

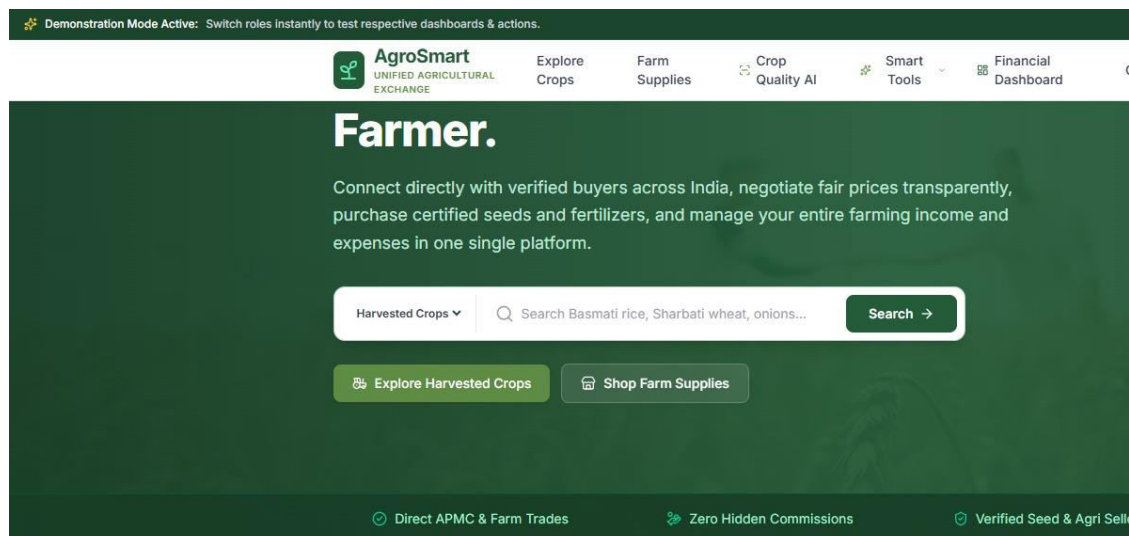


Figure 7: AgroSmart Home Page with Role Switcher, Navigation and Crop Search

Figure 7 shows the home page of the AgroSmart (Unified Agricultural Exchange) platform. A banner indicates demonstration mode with role switching. The navigation bar gives access to Explore Crops, Farm Supplies, Crop Quality AI, Smart Tools and the Financial Dashboard. The search bar has a category selector (set here to Harvested Crops), and the buttons Explore Harvested Crops and Shop Farm Supplies lead to the crop marketplace and the supplies store.

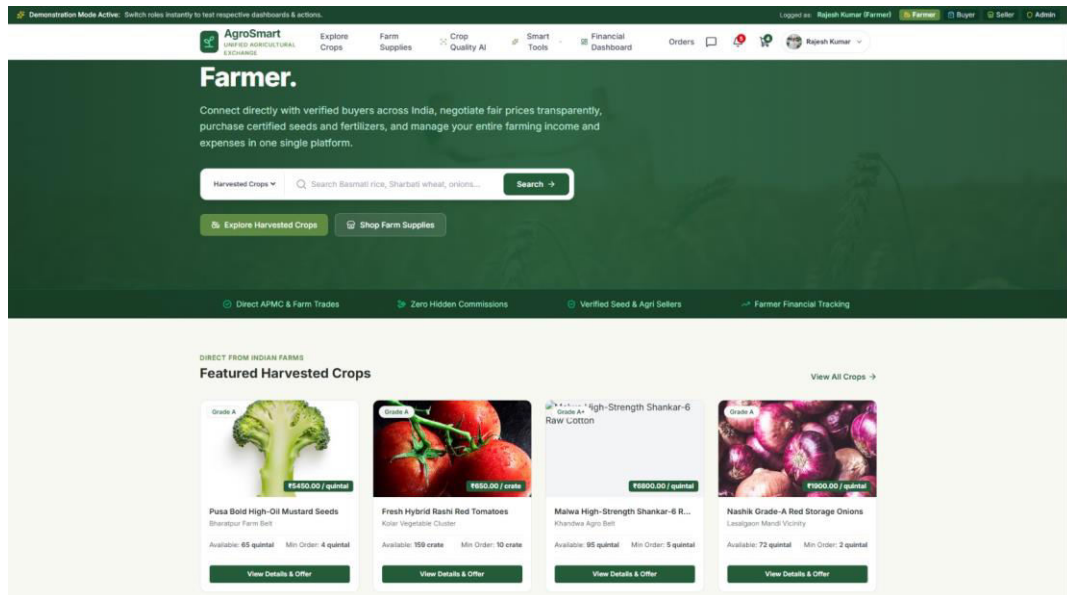


Figure 8: Featured Harvested Crops with Price, Availability and Offer Option

Figure 8 shows the home page for a logged-in farmer. The header provides links to orders, messages, notifications and the cart. The Featured Harvested Crops section displays listing cards with the grade, the price per unit, the available quantity, the minimum order quantity, the product name and the source location, together with a View Details & Offer button that opens the listing for price negotiation.

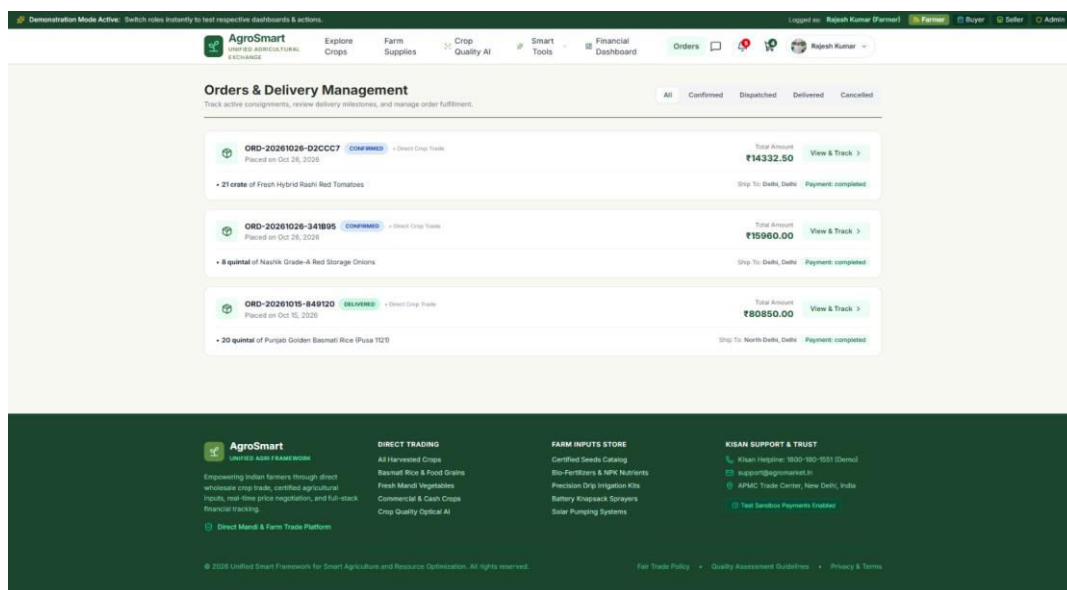


Figure 9: Orders and Delivery Management Interface

Figure 9 shows the Orders & Delivery Management page. Each order is listed with its order number, status (for example Confirmed or Delivered), order type (Direct Crop Trade), date, items and quantity, total amount, delivery location and payment status. Tabs allow the list to be filtered by All, Confirmed, Dispatched, Delivered and Cancelled orders, and the View & Track button opens the details of an order. The footer indicates that payments run in test sandbox mode.

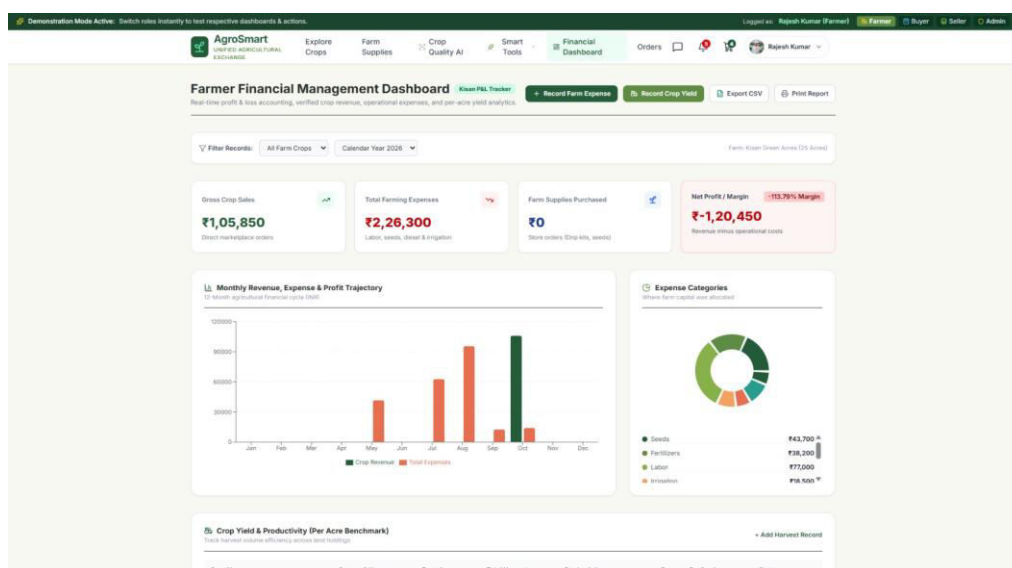


Figure 10: Farmer Financial Management Dashboard

Figure 10 shows the Farmer Financial Management Dashboard. Summary cards present gross crop sales, total farming expenses, farm supplies purchased and net profit or margin, and the records can be filtered by crop and calendar year. A bar chart shows monthly revenue and expenses, a doughnut chart shows expense categories, and a table below tracks crop yield and productivity per acre. The buttons Record Farm Expense, Record Crop Yield, Export CSV and Print Report support data entry and reporting.

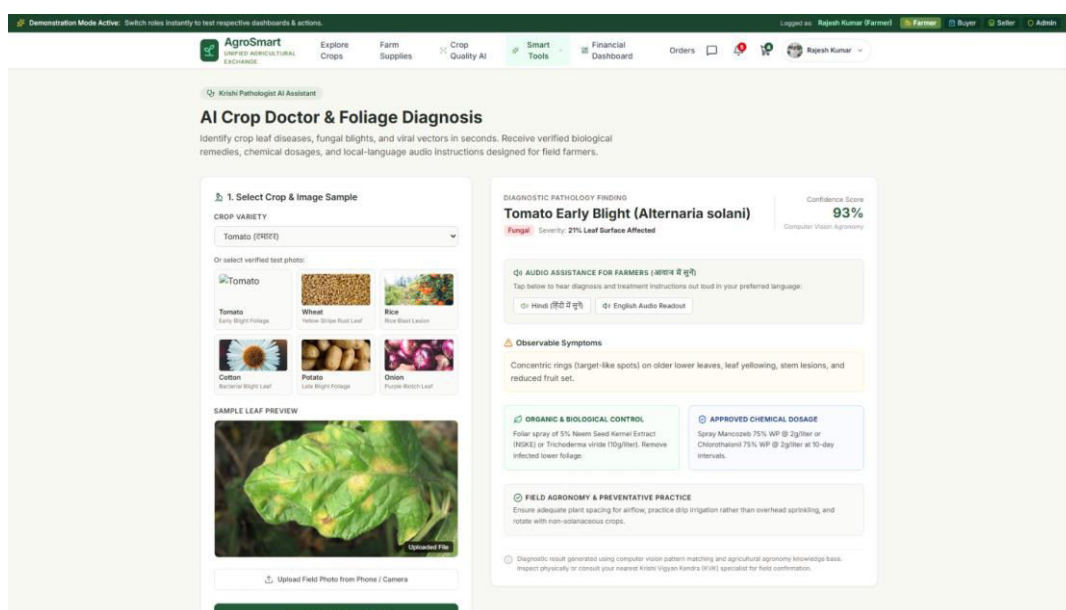


Figure 11: AI Crop Doctor and Foliage Diagnosis Interface

Figure 11 shows the AI Crop Doctor page. The farmer selects a crop and either chooses a sample photograph or uploads a field photograph, and the system returns a diagnosis with a confidence score and severity, an audio readout in Hindi and English, the observed symptoms, organic and chemical control suggestions and preventive field practices. The page notes that the result is generated by pattern matching and should be confirmed with a local agricultural specialist.

VII. CONCLUSION

In conclusion, the development of AgroSmart provides an integrated, role-based and low-friction solution for direct agricultural trading and farm management. By combining a crop marketplace with structured price negotiation, a farm supplies store, order and delivery management, a financial dashboard and an AI Crop Doctor, the platform brings together functions that farmers otherwise handle through several disconnected channels.

The layered architecture based on React, FastAPI and PostgreSQL, together with token-based authentication, role-based access control and atomic order processing, keeps the platform secure and its order, payment and stock records consistent. The recorded offers, messaging and visible order status respond to the finding in the literature that the benefit of digital markets depends on trust and information, while the financial dashboard and the diagnosis module give farmers practical support for planning and crop care.

The project has been completed as a fully working system, and its main modules are demonstrated through the screenshots in this paper. Payments currently run through a sandbox gateway, and the extensions listed under Future Scope can build on the present implementation. Overall, AgroSmart serves as a scalable foundation for smart agriculture, resource optimization and digital market access for farmers.

REFERENCES

- [1] A. Argade, A. K. Laha, and A. K. Jaiswal, "Electronic marketplaces under conditions of oligopsony and relational marketing: An empirical exploration of electronic agricultural markets in India," *Electronic Markets*, 2023, doi: 10.1007/s12525-023-00636-5.
- [2] G. M. B. C. do Prado, E. Catapan, C. M. S. Zanuzzi, F. Matos, and P. M. Selig, "Exploring the key success factors: A case study of a digital marketplace platform for Brazilian small farmers," *Procedia Computer Science*, vol. 232, pp. 159-168, 2024, doi: 10.1016/j.procs.2024.01.016.
- [3] T. Akter et al., "A comprehensive review of external quality measurements of fruits and vegetables using nondestructive sensing technologies," *Journal of Agriculture and Food Research*, art. 101068, 2024, doi: 10.1016/j.jafr.2024.101068.
- [4] R. Vignati et al., "Hyperspectral imaging for fresh-cut fruit and vegetable quality assessment: Basic concepts and applications," *Applied Sciences*, vol. 13, no. 17, art. 9740, 2023, doi: 10.3390/app13179740.
- [5] T. Haerder and A. Reuter, "Principles of transaction-oriented database recovery," *ACM Computing Surveys*, vol. 15, no. 4, pp. 287-317, 1983.
- [6] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *IEEE Computer*, vol. 29, no. 2, pp. 38-47, 1996.
- [7] R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, University of California, Irvine, 2000.
- [8] M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token (JWT)," RFC 7519, Internet Engineering Task Force, 2015.
- [9] "FastAPI documentation." [Online]. Available: <https://fastapi.tiangolo.com>
- [10] "React documentation." [Online]. Available: <https://react.dev>
- [11] "PostgreSQL documentation." [Online]. Available: <https://www.postgresql.org/docs/>



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details