Green Genie: A Smart Farming Platform Using AI and Mobile Technology

Mayur Chavan[1], Atharv Ingale[2], Harshada Avhad[3], Varsha Bombe[4], Aayush More[5], Swati Firame[6]
[1,2,3,4,5]*Sinhgad Institute of Technology and Science, PuneNarhe, Pune*
[6]*Guide, Sinhgad Institute of Technology and Science, PuneNarhe, Pune*

***Abstract*—Green Genie is an innovative smart farming platform developed to address the challenges faced by farmers due to fragmented agricultural resources and lack of timely information. The system aims to improve agricultural productivity and decision-making by providing an integrated digital environment where users can detect crop diseases, receive AI-based crop recommendations, access real-time weather updates, and participate in a digital marketplace for buying, selling, and renting agricultural products. The platform tackles major issues such as limited awareness of modern farming techniques, inaccurate crop selection, and lack of fair pricing in markets. By integrating Artificial Intelligence, mobile technology, and cloud services, Green Genie delivers a structured, user-friendly, and efficient solution tailored for farmers with varying levels of digital literacy. The system is developed using Flutter for the frontend and Python for backend processing, with Firebase used for secure data storage and real-time services. Each user farmer, buyer, and administrator is provided with a role-based dashboard offering customized features and functionalities. The modern and intuitive interface ensures ease of use and accessibility across multiple devices. Green Genie not only enhances communication between stakeholders but also promotes sustainable agricultural practices through intelligent insights and automation. By improving access to information, reducing manual effort, and enabling smart decision-making, the system contributes toward the vision of digital agriculture and farmer empowerment. This project demonstrates how advanced technologies can be effectively utilized to transform traditional farming into a more efficient, scalable, and data-driven ecosystem.**

***Index Terms*—Green Genie, Smart Farming, Artificial Intelligence, Crop Disease Detection, Flutter, Firebase, Digital Agriculture, Marketplace, Farmer Empowerment, Sustainable Farming Smart Farming, Artificial Intelligence, Flutter, Firebase, Crop Detection, Agriculture Technology**

I. INTRODUCTION

In an era of rapid digital transformation, which has significantly improved efficiency in sectors such as education, healthcare, and finance, the agricultural sector still faces numerous challenges, particularly in developing countries. Farmers often struggle with limited access to real-time information, modern farming techniques, and reliable market data. A lack of awareness regarding crop selection, disease identification, and weather conditions creates a substantial gap between traditional farming practices and modern agricultural advancements. This results in reduced productivity, financial losses, and inefficient resource utilization, especially in rural areas where technological adoption remains limited.

Farmers frequently rely on traditional methods, intermediaries, or unverified sources of information, which can lead to incorrect decisions regarding crop management and market transactions. Although various government initiatives and agricultural services exist, the absence of a centralized and integrated digital platform makes it difficult for farmers to access these resources efficiently. This highlights the critical need for a smart, user-friendly, and unified system that can bridge this gap and provide accurate, real-time agricultural support.

The Green Genie project has been developed to address these challenges. It is a smart farming platform that integrates Artificial Intelligence, mobile technology, and cloud services to provide comprehensive agricultural solutions. The system enables users to detect crop diseases through image-based analysis, receive AI-driven crop recommendations, access real-time weather updates, and participate in a digital marketplace for buying, selling, and renting agricultural products. The primary objective of the platform is to enhance decision-making, improve productivity, and empower farmers with modern technological tools.

Developed using Flutter for the frontend and Python for backend processing, Green Genie offers a responsive, cross platform mobile application with a secure and scalable infrastructure supported by Firebase. The system follows a role-based architecture, providing customized dashboards for farmers, buyers, and administrators. This modular approach ensures that all stakeholders can interact efficiently within a single integrated platform.

Green Genie introduces a transformative approach to smart agriculture by combining automation, AI-based insights, and real-time data access. By enabling digital connectivity, it reduces the dependency on manual processes and improves the accuracy of farming decisions. The platform also acts as an advisory system, guiding users with intelligent recommendations and timely alerts, thereby minimizing risks associated with crop failure and market fluctuations.

The motivation behind Green Genie arises from the growing need to modernize agriculture and make technology accessible to farmers with varying levels of digital literacy. This project aims to create an intuitive and reliable platform that simplifies complex agricultural processes. It supports essential functionalities such as user authentication, crop analysis, weather monitoring, and marketplace interactions, all integrated into a seamless workflow. From a technical perspective, the system emphasizes security, usability, and scalability, ensuring that user data is protected while maintaining high performance and future adaptability.

II. LITERATURE REVIEW

The evolution of digital agriculture and smart farming technologies has gained significant attention in recent years. However, accessibility to integrated agricultural solutions remains a major concern, particularly in developing regions. Many existing agricultural applications provide isolated services such as weather forecasting, crop recommendation, or disease detection, but lack a unified and user-friendly platform. This section reviews existing literature on smart farming systems, highlighting their methodologies, contributions, and limitations.

A. The Problem: Fragmentation in Smart Agriculture Systems

Existing studies highlight a significant gap in the adoption of agricultural technologies, particularly among small and marginal farmers. The current farming ecosystem is highly fragmented, with different tools and platforms addressing isolated problems such as weather prediction, crop disease detection, and market pricing. As a result, farmers are required to rely on multiple disconnected systems to gather necessary information. This lack of integration creates challenges such as limited access to real-time and reliable data, low awareness of modern farming techniques, and difficulty in effectively utilizing digital platforms. Consequently, many farmers continue to depend on traditional practices or unverified sources of information, which often leads to poor decision-making, reduced productivity, and financial losses.

B. Review of Existing Solutions: Smart Farming Technologies

The smart agriculture domain has gained significant attention in recent years; however, existing solutions exhibit limitations in addressing the issue of system fragmentation. Current technologies generally fall into two major categories. The first category includes high-complexity, AI-based analytical systems that utilize machine learning models for tasks such as crop disease detection and yield prediction. While these systems offer high accuracy, they are often expensive, resource-intensive, and primarily designed for large-scale agricultural enterprises, making them less accessible to small and marginal farmers. The second category consists of standalone agricultural applications, such as weather forecasting apps and market information portals. Although these platforms improve access to specific services, they operate in isolation and lack integration with other essential agricultural functionalities. As a result, farmers are required to use multiple disconnected platforms, placing the burden of system integration on the end-user and reducing overall efficiency and usability.

C. The Identified Research Gap: Lack of Integrated Smart Farming Ecosystem

The existing literature highlights a clear disconnect between the availability of advanced agricultural technologies and their practical usability by farmers, emphasizing the need for a more holistic and integrated approach. Although numerous solutions have been developed for specific agricultural tasks such as crop disease detection, weather forecasting, and market analysis, these systems operate independently and fail to provide a unified user experience. As a result, a significant research gap exists in the development of a pragmatic, low-cost, and fully integrated smart farming platform. Current studies lack validated models that combine multiple agricultural services into a single cohesive system that can effectively bridge the gap between farmers, agricultural data sources, marketplaces, and advisory services. Addressing this gap is essential to improve accessibility, usability, and overall efficiency in modern farming practices.

D. The Green Genie Contribution

The Green Genie project is designed to address the identified gap in the agricultural ecosystem by providing a unified and integrated smart farming platform. Its primary contribution is not the development of a new standalone algorithm, but the creation of a practical and scalable system architecture that combines multiple agricultural services into a single cohesive application. By approaching the problem as one of integration rather than isolated functionality, Green Genie offers a resource-efficient and user-centric solution. The platform integrates crop disease detection, crop recommendation, real-time weather updates, and a digital marketplace within a single environment, ensuring seamless interaction between different components. Utilizing technologies such as Flutter for cross-platform accessibility and Python for backend intelligence, the system delivers a reliable and easy-to-use tool that empowers farmers with actionable insights. This integrated approach enhances accessibility, reduces dependency on multiple platforms, and promotes efficient and data-driven agricultural practices.

E. The "Human-in-the-Loop" vs. Full Automation Dilemma Legal Tech

literature often focuses on AI and automation, splitting between two poles. On one end are "robot lawyers" and chatbots, an approach fraught with significant ethical, regulatory, and trust-related hurdles, On the other are static, "Web 1.0" informational portals, which are little more than digital phone books. The identified gap is the lack of models for facilitated human connection. The literature is thin on systems that use technology not to replace professionals but to improve the efficiency of the connection to them. The LawBridge project addresses this gap directly. Its architecture is not an "AI lawyer"; it is a "Human-in-the-Loop" facilitator. It operates on the principle that the core problem is a lack of a bridge between services, thus solving the connection problem.

F. Socio-Technical Barriers and Digital Literacy

Beyond economic costs, digital literacy is a critical A2J barrier. The populations most in need often face challenges with internet access, hardware, and the competency to navigate complex online systems. Many modern web platforms can inadvertently increase this barrier. The identified gap is a failure to reconcile advanced technology with the practical, on-the-ground needs of the end-user. The LawBridge project's deliberate selection of a classical, client-side JavaFX framework is a direct response. This design prioritizes stability, a simple UI, and potential for "kiosk-style" deployment. This "resource-aware" philosophy extends to the user's cognitive load, proposing a solution that is less intimidating than complex, cloud-native alternatives.

G. The Failure of "Market-Based" vs. "Public-Good" Models The literature bifurcates the LegalTech ecosystem into two non-collaborating models: "market-based" (commercial) and "public-good" (non-profit/governmental). Commercial models focus on profitable platforms, such as for-profit lawyer directories. These models inevitably focus on high-value, fee generating areas of law and ignore the low-income civil needs that constitute the core of the A2J gap. Conversely, "public good" models (e.g., government portals) are often underfunded, poorly maintained, and lack the technical sophistication of their commercial counterparts.

III. PROBLEM STATEMENT

Agriculture remains a critical sector in developing economies, yet farmers continue to face significant challenges that create a substantial gap between traditional farming practices and modern technological advancements. This gap is not only economic but also structural, arising from the fragmented nature of the agricultural support ecosystem. Farmers often lack access to timely and reliable information regarding weather conditions, crop health, market prices, and suitable farming practices, which directly impacts productivity and income.

Currently, the agricultural ecosystem consists of multiple disconnected systems and services. Farmers must rely on separate platforms for weather updates, crop disease detection, and market information, with no unified solution that integrates these functionalities. Additionally, access to agricultural experts, government schemes, and marketplaces is often limited or poorly coordinated. This lack of integration creates confusion and inefficiency, especially for small and marginal farmers who may have limited technical knowledge.

As a result, farmers are forced to depend on traditional methods, intermediaries, or unverified sources of information. This leads to poor decision-making, increased risk of crop failure, and financial losses. The burden of managing and interpreting fragmented information lies entirely on the farmer, who is not equipped to effectively combine these inputs into actionable insights.

Although various agricultural technologies and applications exist, they fail to address the need for a unified and user-friendly platform. Existing solutions are either complex, costly, and designed for large-scale operations, or they provide limited functionality without integration. Therefore, there is a clear need for a practical, centralized, and intelligent system that can bridge these gaps. Such a solution should integrate multiple agricultural services into a single platform, enabling farmers to access accurate information, make informed decisions, and improve overall productivity and sustainability.

IV. SYSTEM DESIGN AND METHODOLOGY

The Green Genie system was designed and implemented using a structured and systematic methodology to ensure scalability, reliability, and usability. The development process followed a stepwise approach, starting from requirement analysis and system design to implementation, testing, and deployment. This organized methodology ensured that the system meets user requirements while maintaining performance and efficiency. The system architecture is based on a three-tier design consisting of the presentation layer, business logic layer, and data layer, which promotes modularity, flexibility, and ease of maintenance. The presentation layer is developed using Flutter to provide a responsive and user-friendly interface, the business logic layer is implemented using Python for processing and decision-making, and the data layer utilizes Firebase for secure and scalable data storage.

A. System Overview

The system follows a three-layer architecture:

- The Green Genie system is a unified smart farming platform designed to connect farmers, buyers, and administrators for efficient agricultural management.
- It provides an integrated environment where users can access services such as crop disease detection, crop recommendation, real-time weather updates, and a digital marketplace.
- The system follows a modular client-server architecture, with dedicated interfaces for each user role to ensure smooth interaction and usability.
- The technologies used include Flutter (front-end), Python (backend logic), and Firebase (database and real-time services).
- The core objectives of the system include high usability, robust security, scalability, and efficient agricultural data management.

B. System Architecture

- The Presentation Layer handles user interaction. The Flutter-based mobile interface provides role-specific dashboards for farmers, buyers, and administrators, ensuring a responsive and user-friendly experience.
- The Business Logic Layer manages system operations and processing. Python-based backend services handle functionalities such as user authentication, crop disease detection, crop recommendation, weather data processing, and marketplace operations, along with validation and error handling.
- The Data Layer (Firebase Firestore) stores and manages application data, including user details, crop information, transactions, and real-time

updates, ensuring scalability and efficient data retrieval.
- Secure communication between the frontend and backend is maintained using APIs, and Firebase services ensure authentication, data security, and real-time synchronization.

C. Methodology
- The Requirement Analysis phase involved identifying functional requirements such as user registration, crop disease detection, crop recommendation, weather updates, and marketplace operations, along with non-functional requirements like usability, scalability, performance, and security.
- In the System Design phase, system models such as DFDs, ER diagrams, and workflow diagrams were prepared, and the Firebase database structure was defined for efficient data storage and retrieval.
- During the Implementation phase, the Flutter-based user interface was developed, and backend logic was implemented using Python, following a modular and structured approach for maintainability.
- API integration and database connectivity were established using Firebase services, ensuring secure data handling, authentication, and real-time synchronization.
- The Testing phase included unit testing, integration testing, and usability testing to ensure system functionality and performance. Security testing was also conducted to validate data protection and access control mechanisms.
- The Deployment phase involved configuring Firebase services, deploying backend APIs, and publishing the mobile application for end-users with secure credential management.
- The Evaluation and Maintenance phase focused on performance monitoring, user feedback collection, and continuous improvements to enhance system efficiency, scalability, and user experience.

D. Validation and Evaluation
- Functional validation ensured that all modules, including user authentication, crop disease detection, crop recommendation, weather updates, and marketplace operations, met the requirements defined in the system specification.
- Usability testing confirmed that the mobile interface developed using Flutter is user-friendly, with intuitive navigation, clear design, and smooth performance across different devices.
- Security validation ensured proper implementation of authentication mechanisms, secure data storage using Firebase, and protection of user data through access control and encryption techniques.
- Performance metrics such as response time, API latency, and data retrieval speed were analyzed to verify the system's ability to handle multiple user requests efficiently.
- The system was continuously monitored, and maintenance procedures were established for regular updates, performance optimization, and secure data backups to ensure long-term reliability.

V. EXPERIMENTATION AND IMPLEMENTATION

1) Implementation Process:
- The implementation began with setting up the development environment and initial configuration. Flutter SDK and required tools were installed for frontend development, while Python frameworks (Flask/Fast-API) were configured for backend processing. Firebase was initialized with collections for users, crops, transactions, and system data.
- Database connectivity and backend integration were established using Firebase services and REST APIs. Secure communication was ensured through authentication mechanisms, and structured data handling was implemented for efficient storage and retrieval.
- The frontend design was developed using Flutter, with screens such as Login, Registration, Dashboard, Crop Analysis, and Marketplace. The UI followed a modern design approach with consistent color schemes, responsive layouts, and user-friendly navigation to enhance accessibility.
- The controller and backend logic handled all system operations and user interactions. Python-based services processed user requests, managed data validation, handled API calls (weather and market data), and ensured smooth communication between components. Error handling mechanisms were implemented to maintain system stability.

- Core functionalities were implemented, including user authentication, crop disease detection using CNN models, crop recommendation systems, real-time weather integration, and digital marketplace features. Secure authentication and data handling ensured that sensitive user information was protected.
- The crop disease detection module was developed using an AI model that analyzes uploaded images and predicts possible diseases, providing recommendations for treatment and prevention.
- After implementing all modules, the testing phase was conducted. Unit testing verified individual components, while integration testing ensured proper interaction between frontend, backend, and database systems. The system was also tested under multiple user scenarios to evaluate performance and reliability.

2) Experimentation and Analysis

- Experimentation was conducted in a controlled environment to evaluate the performance and reliability of the Green Genie platform. The testing setup included a local system with standard hardware configuration, Firebase as the cloud database, and Python-based backend services. Different user roles farmer, buyer, and administrator were tested under various input scenarios to analyze system functionality and data flow.
- Initial experiments focused on measuring response time. The system's average response time for user login and dashboard loading was approximately 1 second, while crop disease detection results were generated within 2–3 seconds depending on image size and processing complexity. Marketplace operations and data retrieval were completed within acceptable performance limits.
- Database performance was evaluated by executing multiple read and write operations simultaneously. Firebase's real-time database capabilities ensured efficient data synchronization, and optimized queries improved retrieval speed. The system demonstrated the ability to handle multiple concurrent users without noticeable performance degradation.
- Security testing was conducted to ensure protection against unauthorized access and data breaches. Authentication mechanisms, secure API communication, and Firebase security rules prevented unauthorized data manipulation and ensured safe handling of user information.
- Usability testing was performed with a group of users, including students and individuals familiar with basic smartphone usage. Participants were asked to perform tasks such as registration, crop image upload, and accessing marketplace features. Feedback indicated high satisfaction in terms of ease of use, interface clarity, and navigation.
- The crop disease detection module was evaluated based on accuracy and response time. Experimental results showed that the AI model provided reliable predictions for common crop diseases, assisting users in taking appropriate actions.
- The weather and market data modules were tested for real-time updates and accuracy. The system successfully displayed updated information without delays, validating the integration of external APIs.
- The system was analyzed under concurrent usage scenarios to ensure stability. It maintained consistent performance with multiple active users, and resource utilization remained within acceptable limits, ensuring scalability for future expansion.

VI. RESULTS AND DISCUSSION

The implementation and experimentation of the Green Genie system produced positive and reliable results. The primary outcome is a fully functional smart farming application that successfully integrates key stakeholders, including farmers, buyers, and administrators, into a unified platform. Functional validation through testing confirmed that all core system requirements were successfully achieved, including user authentication, crop disease detection, crop recommendation, real-time weather updates, and digital marketplace operations.

The system demonstrated efficient performance in terms of response time and data processing. Crop disease detection using the AI model provided accurate and timely results, assisting farmers in identifying issues and taking corrective actions. The integration of real-time weather and market data enhanced decision-making by providing users with up-to-date information. The marketplace module enabled smooth transactions between farmers and buyers, improving accessibility and reducing dependency on intermediaries.

Usability testing indicated that the application is intuitive and easy to use, even for users with basic digital knowledge. The Flutter-based interface ensured responsiveness across devices, while the backend system maintained reliable communication and data handling. Performance analysis showed that the system can handle multiple concurrent users without significant degradation, demonstrating its scalability.

Overall, the results confirm that Green Genie effectively addresses the challenges of fragmented agricultural systems by providing a centralized, intelligent, and user-friendly platform. The discussion highlights that integrating AI with realtime data and a digital marketplace significantly improves efficiency, productivity, and user engagement in modern agriculture.

A. Successful Implementation of Role-Based Access Control (RBAC)

- The authentication module of the Green Genie system is fully functional and securely implemented.
- Valid users are correctly authenticated and redirected to their respective dashboards based on roles (e.g., Farmer to Farmer Dashboard, Buyer to Buyer Dashboard, Administrator to Admin Dashboard).
- Invalid login attempts are denied with appropriate error messages, ensuring protection against unauthorized access.
- Firebase Authentication and secure backend validation ensure that user credentials are handled safely and efficiently.
- This confirms the system's strong security foundation and effective role-based access control, successfully distinguishing between different user types and their respective functionalities.

B. Proven Data Integration and "Bridge" Functionality

- This represents a critical outcome of the Green Genie system, validating its core concept of integrating multiple agricultural services into a single unified platform.
- The system successfully captures and stores user data, crop details, and transaction records within the Firebase database, ensuring reliable data persistence.
- The crop disease detection module effectively processes uploaded images and returns accurate predictions along with recommendations, demonstrating seamless integration between the frontend, backend, and AI model.
- The weather and market data modules correctly fetch and display real-time information from external APIs, ensuring users receive up-to-date and relevant data.
- The digital marketplace module allows farmers and buyers to interact efficiently, with product listings and transactions being dynamically updated and reflected across user interfaces.
- Overall, the system demonstrates a functional proof of concept of a unified smart farming platform, successfully bridging the gap between multiple agricultural services and eliminating the need for separate, disconnected systems.

C. System Performance and Responsiveness

- The system performance of Green Genie was evaluated to validate the efficiency of its lightweight and scalable technology stack.
- Data retrieval operations from Firebase, including user data, crop details, and marketplace listings, were executed efficiently, with average response times typically under 200 milliseconds for standard queries. Crop disease detection results were generated within 2–3 seconds, depending on image size and processing complexity.
- The Flutter-based user interface was smooth, responsive, and user-friendly. Screen transitions, navigation between modules, and user interactions such as image uploads and marketplace browsing were performed without noticeable lag.
- Real-time data synchronization ensured that updates in weather information, crop analysis results, and market place activities were reflected instantly across the application.
- These results confirm that the chosen technology stack provides a high-performance and reliable system, capable of delivering efficient smart farming solutions without requiring overly complex infrastructure.

VII. CONCLUSION

This project successfully addresses the critical issue of fragmentation within the agricultural ecosystem by developing a unified and intelligent smart farming platform. The central idea of this work is that the challenges faced by farmers are not only due to lack of resources, but also due to the absence of an integrated system that combines multiple agricultural services into a single accessible solution. The primary objective was to design, implement, and validate a technological platform that bridges the gap between farmers, data sources, and market services.

The outcome of this project is a fully functional and high-performance application that effectively achieves this objective. By utilizing a modern and scalable technology stack including Flutter, Python, and Firebase, the Green Genie system provides a seamless and responsive platform where users can access crop disease detection, crop recommendations, real-time weather updates, and marketplace functionalities within a single interface. Experimental results confirm that the system meets all core functional requirements while maintaining efficiency, usability, and reliability.

The primary contribution of this work lies not in the development of a new standalone algorithm, but in the creation of a practical and integrated system architecture for smart agriculture. It addresses a significant gap in existing solutions by providing a unified platform that combines multiple services, thereby improving accessibility, reducing dependency on fragmented tools, and enabling data-driven decision-making for farmers.

The system performs well but has limitations in scalability and AI accuracy. Future improvements include IoT integration, better machine learning models, and a more scalable cloud infrastructure. Overall, Green Genie enhances agricultural efficiency through a smart and user-friendly solution.

REFERENCES

[1] Akkem Y, Biswas SK, Varanasi A. Smart farming using artificial intelligence: A review. Eng Appl Artif Intell. 2023.

[2] Nitin, Gupta SB. Artificial intelligence in smart agriculture: Applications and challenges. Curr Appl Sci Technol. 2023;24(2).

[3] Memon K, et al. Artificial intelligence in sustainable smart agriculture: Concepts, applications and challenges. VAWKUM Trans Comput Sci. 2025.