IJIRAE:: International Journal of Innovative Research in Advanced Engineering ISSN: 2349-2163
Volume 12, Issue 11, November 2025 https://www.ijirae.com/archives
https://doi.org/10.26562/ijirae.2025.v1211.27

AgriAid: AI-Powered Agricultural Assistant for Inclusive Plant Disease Detection

Prof.Anjali Gajjar
Assistant Professor, Department of CSE-DS,
AMC Engineering College, Bengaluru, India
gajjaranjali224@gmail.com
https://orcid.org/0000-0001-6987-2657
Tulsi Kumar Yadav, Vanteru Mounika, Shraddha S Methri, Shivam Kumar
Department of CSE-DS
AMC Engineering College Bengaluru, India
1am22cd107@amceducation.in, 1am22cd109@amceducation.in
1am22cd091@amceducation.in, 1am22cd090@amceducation.in

Publication History
Manuscript Reference No: IJIRAE/RS/Vol.12/Issue11/NVAE10106
Research Article| Open Access | Double-Blind Peer-Reviewed | Article ID: IJIRAE/RS/Vol.12/Issue11/NVAE10106
Received: 01, November 2025, Revised: 08, November 2025, Accepted:20, November 2025, Published Online: 30, November 2025. https://www.ijirae.com/volumes/Vol12/iss-11/27.NVAE10106.pdf
Citation: Prof.Anjali,Tulsi,Vanteru,Shraddha,Shivam (2025),AgriAid: AI-Powered Agricultural Assistant for Inclusive Plant Disease Detection, IJIRAE: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 11 of 2025 pages 609-614 Doi:>https://doi.org/10.26562/ijirae.2025.v1211.27
BibTeX Key: Prof.Anjali@2025AgriAid
IJIRAE papers should be cited as IJIRAE (International Journal of Innovative Research in Advanced Engineering, AM Publications, India 2025, ISSN 2349-2163, https://doi.org/10.26562/ijirae.2025.v1211.27 The journal's official abbreviation is IJIRAE. Orcid: https://orcid.org/0009-0004-9398-7488

Abstract: Agriculture remains the backbone of the economy, yet farmers frequently grapple with yield- limiting challenges such as unidentified plant diseases, unpredictable weather patterns, and volatile marketprices.Whilerecentadvancementsindeeplearninghavedemonstratedhighaccuracyindisease classification under laboratory conditions [1], many existing solutions fail to address real-world constraints such as linguistic barriers, low literacy, and intermittent internet connectivity. This paper presents AgriAid, an inclusive, AI-powered mobile ecosystem designed to democratize access to agricultural expertise. The proposed system employs a multimodal input framework accepting voice, text, and image data to facilitate intuitive interaction for users across diverse demographics. Leveraging Convolutional Neural Networks (CNN) for high-precision image-based disease detection and the Google Gemini API for generative, context-aware agricultural advisory, the platform provides real-time diagnosis, remedial measures, and personalized crop recommendations. Furthermore, the application integrates hyper local 7-day weather forecasts, government scheme eligibility checks, and live market price tracking into a multilingual interface. Developed using React Native and Node.js, the system is optimized for offline capability, ensuring robust decision support even in remote, low-connectivity environments.

Keywords: Smart Farming, Plant Disease Detection, Generative AI, Multimodal Interaction, CNN, Google Gemini API, React Native, Agricultural Intelligence.

I. INTRODUCTION

Agriculture is the primary source of livelihood for approximately 58% of India's population, serving as the backbone of the nation's economy. Despite its critical importance, the sector faces multifaceted challenges, including unpredictable climate variability, fragmented landholdings, and a lack of real-time access to agronomic information. A significant portion of crop yield is lost annually due to unidentified pests and diseases, often exacerbated by the farmers' inability to diagnose symptoms early and accurately. While the proliferation of smart phones and affordable internet in rural India has opened new avenues for digital agriculture, a substantial "digital divide" remains. Existing technological solutions often alienate the very users they aim to serve due to complex text-based interfaces, language barriers.

Recent advancements in Artificial Intelligence (AI) and Computer Vision have shown immense potential in addressing these issues. Deep learning models, particularly Convolutional Neural Networks (CNNs), have demonstrated remarkable accuracy in classifying plant diseases under controlled laboratory conditions [1], [3]. However, the transition from laboratory prototypes to field-deployable tools has been fraught with challenges. Many existing systems are computationally intensive [3], requiring high-end hardware that is unavailable to the average farmer. Others rely heavily on continuous internet connectivity [6], rendering them ineffective in remote low-bandwidth areas. Furthermore, while conversational AI systems and chat bots have been introduced to bridge the literacy gap [6], [7], they often lack "visual intelligence" the ability to process image-based inputs for accurate diagnosis.

Conversely, image-based detection apps [4], [5] typically lack the interactive, advisory capacity to explain why a disease occurred or how to treat it in a localized context. There is a distinct lack of a holistic ecosystem that combines high-precision visual detection with generative, context-aware advice in a format accessible to semi-literate or illiterate users. To address these gaps, this paper introduces AgriAid, an inclusive, AI-powered agricultural ecosystem designed for the Indian context. AgriAid leverages a multimodal input framework, allowing farmers to interact via voice, text, or image, thereby overcoming literacy barriers. The system integrates a lightweight CNN model for real-time, offline disease detection with the Google Gemini API to generate personalized, context-aware remedies and preventive measures. Additionally, AgriAid incorporates hyper local weather forecasting, government scheme eligibility checks, and real-time market insights into a single, multilingual interface. By prioritizing accessibility and offline capability, AgriAid aims to empower farmers with actionable data-driven insights, ensuring sustainable productivity and economic security.

II. LITERATURE SURVEY

The development of AgriAid is grounded in a comprehensive review of existing methodologies in agricultural machine learning, specifically focusing on image-based disease detection, conversational assistance systems, and the limitations of current technological interventions.

A. Deep Learning in Plant Disease Detection Foundational research has established the efficacy of Convolutional Neural Networks (CNNs) in agriculture. Mohanty et al. [1] proposed a CNN-based system trained on a dataset of over 54,000 images, achieving high classification accuracy under controlled laboratory conditions. However, the study highlighted a significant performance drop when applied to real-world scenarios due to variations in lighting and background, a limitation AgriAid addresses through robust preprocessing. Similarly, Sladojevic et al. [2] developed a precision CNN model for 13 distinct plant diseases. While highly accurate, the system was restricted to desktop environments, lacking the mobile compatibility and offline capabilities required for in-field use by farmers. Further advancements focused on architectural comparisons. Ferentinos [3] analyzed AlexNet, VGG, and Res Net architectures, reporting over 99% accuracy on large datasets. Despite these results, the high computational overhead identified in the study rendered these models unsuitable for deployment on resource-constrained mobile devices found in rural India. To address image quality issues, Barbedo [4] emphasized preprocessing strategies such as histogram equalization to enhance robustness. AgriAid incorporates similar techniques to ensure reliable detection even with low- quality camera inputs. More recent approaches have explored hybrid architectures. Brahimietal. [5] demonstrated the efficiency of transfer learning using VGG16, while Kanakala and Ningappa [8] introduced a hybrid CNN-LSTM model to capture spatial-temporal disease patterns with 96.4% accuracy. Thakur et al. [10] further advanced this field by proposing a Vision Transformer (ViT-CNN) hybrid for explainable diagnosis. However, a common limitation across these studies [5], [8], [10] is the focus on algorithmic accuracy over user accessibility; these systems typically lack localized interfaces or explanations understandable to non-expert users. A comprehensive review by Abade et al. [9] confirmed that while CNNs perform well invitro, a significant gap remains in adapting these models to field conditions with region-specific data.

B. Conversational AI and Assistive Technologies In parallel, research has explored voice-based interfaces to overcome literacy barriers, a key demographic challenge in Indian agriculture. Patel et al. [6] introduced "AgriBot," a speech-enabled assistant for local languages. While accessible, it was limited to text and voice queries and lacked the "visual intelligence"todiagnosecrophealthfromimages.Addressingconnectivity issues, Chen et al. [7] developed an offline voice assistant for low- bandwidth regions. However, this system relied solely on verbal descriptions from the farmer, missing the critical component of image- based validation which allows for more accurate diagnosis. Identification of Research Gap The literature indicates a dichotomy: existing systems are either highly accurate in visual detection but computationally heavy and inaccessible [1], [3], [10], or they are accessible and voice-driven but lack visual diagnostic capabilities [6], [7]. There is a lack of a unified, multimodal ecosystem that combines high- precision visual detection with generative, context-aware advice. AgriAid bridgesthisgapbyintegratingefficientCNN-basedvisualdetectionwitha Generative AI (Gemini) powered conversational interface, delivering a multimodal (Text, Image, Voice), multilingual, and offline-capable solution tailored for the rural Indian context.

III. PROPOSED SYSTEM

The AgriAid platform is designed as a holistic, AI-driven ecosystem that integrates advanced machine learning models with real-time agronomic data to support farmers throughout the crop cycle.

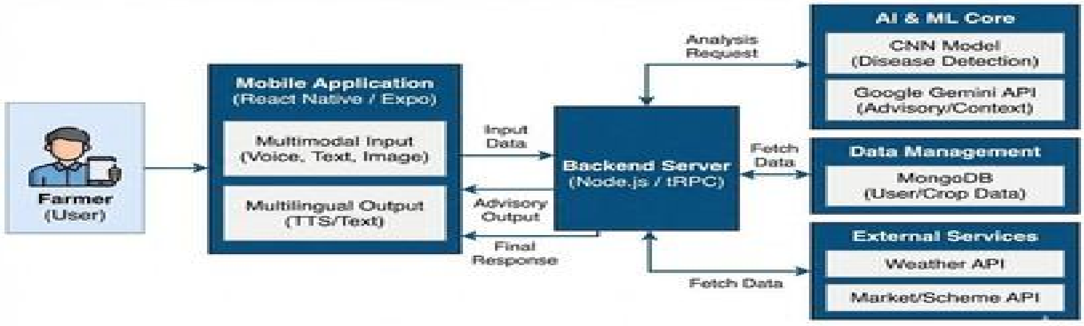

Fig. 1. Proposed AgriAid System Architecture.

Unlike traditional systems that function in isolation (e.g.,only disease detection or only weather forecasting), AgriAid unifies these functionalities into a single, multimodal interface. The proposed system operates on a modular architecture consisting of three primary components: the Multimodal Input Engine, the Core Intelligence Engine, and the Localized Advisory Module Multimodal Input Engine To address the digital divide and literacy barriers prevalent in rural India, the system is designed with a "Voice-First" and "Image-First" approach. Farmers interact with the application through three distinct channels:

Visual Input: Users capture images of crop leaves using the specialized camera module powered by expo-image- picker. Verbal Input: Voice commands in local languages (Kannada, Hindi, English) are captured and processed using expo-av and Speech-to-Text APIs. Text Input: Traditional text queries are supported for literate users or specific data entry.

A. Core Intelligence Engine This module serves as the brain of the system, processing inputs to generate actionable insights:

1. Disease Diagnosis: Image data is fed into a lightweight Convolutional Neural Network (CNN) optimized for mobile devices. This model detects specific plant diseases (e.g., blights, rusts) with high precision.
2. Contextual Reasoning: The system leverages the Google Gemini API to analyze the context of the disease, weather conditions, and user queries. Unlike static database lookups, Gemini generates dynamic, context-aware remedies and preventive measures tailored to the specific situation.

B. Localized Advisory Module The system aggregates external environmental and economic data to provide a comprehensive decision-support framework:

Weather Analytics: Real-time and 7-day forecast data are fetched to provide advisories on irrigation and sowing schedules, helping farmers mitigate climate risks.

Economic Intelligence: The module integrates real-time market prices (Mandi rates) and checks eligibility for government schemes (subsidies, insurance, loans).

C. Unified Output The final insights ranging from disease diagnosis to market trends—are delivered back to the user in their native language via both text and Text-to-Speech (TTS) voice output. This ensures that the information is accessible even to users with limited reading proficiency. Built using React Native with Expo, AgriAid runs seamlessly across Android, iOS, and web platforms, ensuring broad scalability and deployment reach. Ultimately, AgriAid aims to transform plant disease diagnosis by offering a real-time, reliable, and farmer-friendly experience tailored for underserved agricultural communities. By reducing reliance on chemical treatments and mitigating crop losses, it contributes to sustainable agriculture, improved farmer livelihoods, and greater food security.

IV. SYSTEM ARCHITECTURE

The system architecture of AgriAid is designed as a robust, three-tiered client-server model that facilitates seamless dataflow from the user to the AI processing engine and back. The architecture prioritizes low latency, offline accessibility, and multimodal interaction to cater to the specific constraints of rural agricultural environments.

A. User Interface Layer (Client Side) The client side is developed using React Native (SDK 53) and Expo, ensuring a cross-platform experience on both Android and iOS devices. This layer serves as the primary entry point for the farmer, featuring a "Multimodal Input Interface" that accepts:

Voice: Capturing local dialects via expo-av.

Images: capturing high-resolution leaf images via expo- image-picker for disease analysis.

Text: allowing manual query entry. To ensure functionality in low-connectivity regions, the client layer implements an Offline-First strategy using A sync Storage, caching critical data such as weather forecasts and previous diagnosis reports locally on the device.

B. Application Layer (Server Side) the core logic resides in the Application Layer, built on a Node.js runtime environment. This layer utilizes tRPC (Type Script Remote Procedure Call) to establish a type-safe API boundary between the frontend and backend, ensuring reliable data transmission.

Request Orchestration: The server receives user inputs and routes them to the appropriate processing module. Voice inputs are first processed by a Speech-to-Text engine before being analyzed.

Validation: All incoming data is rigorously validated using Zod schemas to prevent malformed queries from reaching the AI models.

C. Intelligence & Data Layer This layer houses the decision- making components of the system:

AI Processing Engine: This module integrates the Google GeminiAPI for generative reasoning and a dedicated CNN model for image-based disease detection. The Gemini API interprets the context of the user's query (e.g., specific symptoms, weather conditions) to generate human-like, actionable advice.

Database: A centralized MongoDB database is employed for persistent storage. It stores user profiles, historical query data, crop details, and government scheme information.

External Integrations: The system connects with external APIs to fetch real-time environmental data (Weather API) and market fluctuations, enriching the AI's analysis before the final response is generated.

D. Output & Feedback Module Once the analysis is complete, the results are processed by the Feedback Module. This component formats the technical diagnosis into simple, vernacular language. The response is delivered via the mobile app in both text and voice formats (using expo- speech), ensuring that the information is comprehensible to users regardless of their literacy level.

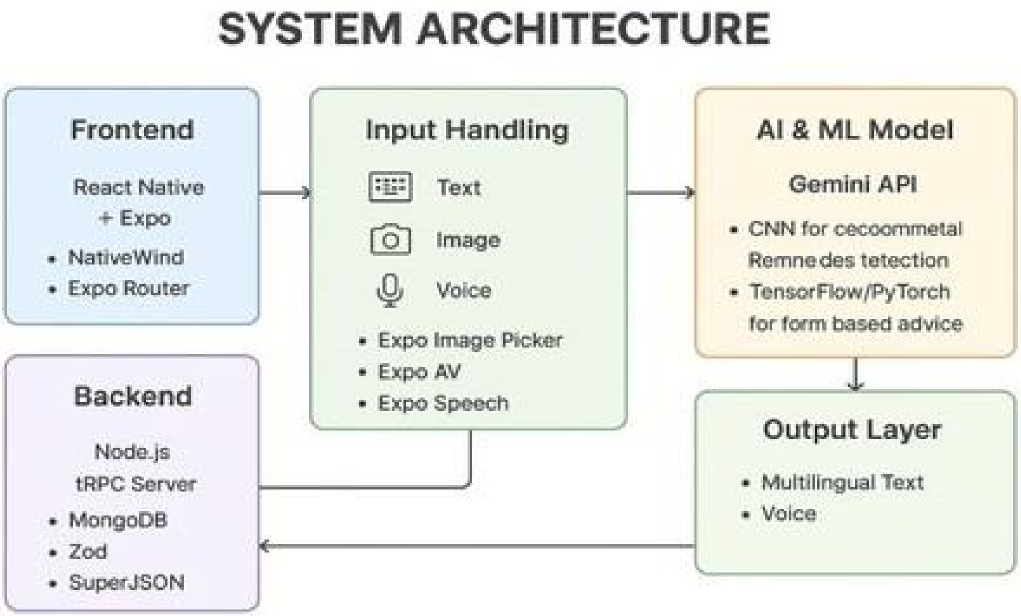

Fig 4.1 System Architecture

V. METHODOLOGY

The proposed system, AgriAid, is an AI-powered multimodal agricultural assistant designed to provide accurate plant disease diagnosis and contextual agricultural guidance to farmers through a cross-platform mobile solution. The architecture integrates advanced frontend technologies, robust backend services, and intelligent machine learning models to deliver real-time, multilingual, and accessible support.

1. Frontend Architecture The user interface is developed using React Native with Expo, ensuring seamless deployment across Android and iOS platforms. The interface design follows accessibility-first principles and is styled using Native Wind, enabling responsive and visually consistent components. Navigation is managed through Expo Router, providing intuitive and efficient screen transitions tailored to users with varying levels of digital literacy. To support offline functionality a critical requirement identified in [7] the frontend implements local caching strategies using Async Storage.
2. Multimodal Input Handling To accommodate the diverse literacy levels of rural farmers, the system processes three distinct data streams: This multimodal approach aligns with the principles of universal design, ensuring that critical agricultural diagnostics are accessible to users regardless of their formal education or technological proficiency. Furthermore, by decoupling the input modality from the processing logic, the system allows farmers to seamlessly switch between voice, text, and image interactions based on their immediate field conditions and comfort level. communication with third-party services such as the Google Gemini API and weather providers. This design abstracts complex API logic from the client and ensures the secure management of sensitive credentials. Output Generation Layer The system delivers comprehensive, accessible outputs through dual channels: Image Input: Implemented through Expo Image Picker, this module enables farmers to capture high-resolution leaf images directly from the field. These images undergo client-side compression before transmission to reduce latency. Voice Input: Powered by Expo AV, this feature allows users to describe symptoms verbally in their native language. This overcomes the limitations of text-only interfaces noted in [6]. Location Services: Supported via Expo Location, this module fetches geotags to contextually filter advice based on region- specific soil data, real-time weather insights, and local market conditions.
3. AI and Machine Learning Model Layer The intelligence layer leverages a hybrid approach combining generative and discriminative AI models:
   Discriminative Diagnosis: A specialized Convolutional Neural Network (CNN) is employed to identify plant diseases from user-submitted images. Consistent with the standards established in [1], this model is optimized for feature extraction to differentiate between healthy and diseased leaf tissue with high accuracy.
   Generative Advisory: The system integrates the Google Gemini API for natural language understanding and reasoning. It synthesizes the CNN's diagnostic output with weather and soil data to generate holistic remedies, preventive measures, and crop recommendations.
   Text Processing: For textual queries, Tensor Flow/Py Torch- based NLP models facilitate the understanding of agricultural jargon and vernacular phrasing.
4. Backend Infrastructure The backend is architected on a Node.js runtime environment, utilizing tRPC (Type Script Remote Procedure Call) to ensure type-safe communication between the frontend and backend.

Database: MongoDB serves as the primary non-relational data store, managing user profiles, diagnosis logs, and agricultural datasets. Validation & Serialization: To ensure data integrity, Zod is utilized for rigorous schema validation at the API boundary, while Super JSON handles the efficient serialization of complex data structures (such as date objects and maps) during transmission.

External Service Integration: The backend functions as a secure middleware layer, orchestrating asynchronous,

Multilingual Text Output: The UI displays diagnosis results, treatment recommendations, and government scheme information in the user's selected local language.

Voice Feedback: Implemented using expo-speech, this module converts textual advice into spoken audio. This feature ensures that the system remains accessible to illiterate users, directly addressing the inclusivity gaps identified in [5].

6. Deployment and Development Support The methodology incorporates modern DevOps practices to ensure scalability:

   Local Testing: Rapid development and debugging are facilitated through Expo's local runtime environment. Remote Demonstrations: Expo Tunnel is employed to enable secure remote access for real-device testing and stakeholder demonstrations without complex network configurations.

VI. IMPLEMENTATION

The implementation of the AgriAid ecosystem prioritizes accessibility, real-time performance, and cross-platform compatibility. The system is engineered using a decoupled architecture, separating the client-side presentation layer from the server-side logic and intelligence engine. This section details the specific technologies, libraries, and frameworks employed to realize the proposed architecture.

A. Frontend Development the mobile application is developed using React Native (SDK53) coupled with the ExpoCLI, enabling a unified codebase for both Android and iOS deployment.

   User Interface (UI): To accommodate users with varying degrees of digital literacy, the UI follows an accessibility-first design philosophy. Native Wind is utilized for utility-first, responsive styling, ensuring consistent rendering across different screen sizes. Navigation: Screen transitions and deep linking are managed by Expo Router, which provides a file-based routing system optimized for performance and maintainability.

B. Multimodal Input Integration To facilitate the "Voice-First" and "Image-First" approach, the system integrates specific Expo SDKs to handle diverse data types:

   Image Acquisition: The expo-image-picker library is implemented to allow farmers to capture real-time images of crop leaves or upload existing photos from the gallery for disease analysis.

   Audio Processing: Voice commands and symptom descriptions are captured using expo-av. The system supports regional dialects by integrating native speech-to-text engines.

   Location Services: expo-location is utilized to fetch geospatial coordinates, which are essential for retrieving localized weather data and checking region-specific government scheme eligibility.

C. AI Core and Algorithmic Processing The intelligence layer functions as a hybrid frame work, combining discriminative models for detection with generative models for advisory:

   CNN-Based Classification: A specialized Convolutional Neural Network (CNN) processes the image inputs. Trained on annotated agricultural datasets, this model identifies specific plant pathologies (e.g., Leaf Rust, Blight) with high precision.

   Generative Reasoning (GeminiAPI): The system integrates the Google Gemini API to augment the CNN's output. Gemini analyzes the detected disease in the context of user inputs (soil type, season) to generate holistic, human-like remedies and preventive measures.

D. Backend Infrastructure The server-side logic is built on a Node.js (v18+) runtime, ensuring non-blocking, event-driven processing.

   API Layer: The system utilizes tRPC (Type Script Remote Procedure Call) to establish end-to-end type safety between the client and server. This reduces runtime errors and streamlines API development.

   Data Management: Mongo DB acts as the scalable, NoSQL document store for user profiles, diagnosis logs, and market data.

   Validation & Serialization: Data integrity is enforced usingZod schemas for strict input validation, while Super JSON is employed to serialize complex data types (such as Dates and Maps) that are typically lost in standard JSON transmission.

E. Output & Accessibility Layer The feedback mechanism ensures that the complex AI analysis is comprehensible to the end-user:

   Voice Feedback: The expo-speech library converts the generated textual advice into spoken audio in the user's preferred regional language, bridging the literacy gap.

   Multilingual Text: A dynamic translation module renders diagnosis results, weather alerts, and market prices in the local vernacular.

F. Deployment and DevOps The development lifecycle incorporates modern tooling for continuous integration and testing: Compilation: Babel and the Metro Bundler are used for efficient JavaScript transpilation and asset management. Code Quality: ESLint is integrated into the workflow to enforce coding standards and prevent syntax errors. Remote Testing: Expo Tunnel (Ngrok) is utilized to enable secure, remote access to the development server, allowing the team to test the application on physical devices in low-connectivity field environments without complex network configuration.

VII. FUTURE SCOPE

AgriAid aims to evolve into a fully autonomous and deeply integrated agricultural ecosystem. Future enhancements focus on reducing dependency on cloud infrastructure, enhancing data granularity, and fostering economic resilience.

A. Edge AI & Offline Inference to eliminate connectivity requirements, future iterations will deploy quantized CNNs and distilled Large Language Models (LLMs) directly on mobile devices. This will enable fully offline disease diagnosis and advisory generation, ensuring functionality in the most remote regions [7].

B. Drone &IoT Integration Scaling from individual plant diagnosis to field-level surveillance, the system will integrate with autonomous drones for aerial spectral imaging [10]. Additionally, support for IoT soil sensors will be added to ingest real-time NPK and moisture data, facilitating precision agriculture and automated irrigation schedules [9].

C. Explainable AI (XAI) & Trust To enhance user confidence in automated diagnosis, Explainable AI (XAI) frameworks will be implemented. By utilizing techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM), the application will visually highlight specific lesion a reason leaf images that triggered the diagnosis. This visual feedback educates farmers and validates the model's decision-making process.

D. Advanced Linguistic Capabilities To further lower entry barriers, the NLU engine will be upgraded to process code-mixed languages (e.g., Hinglish) and diverse regional dialects, ensuring seamless interaction for users with limited formal education [6].

E. Direct Market Linkage & Predictive Supply Chain Future developments will extend beyond advisory services to include a farm-to-market module. By leveraging yield prediction algorithms, the platform will connect farmers directly with buyers and logistics providers prior to harvest. This predictive supply chain integration aims to reduce post-harvest losses and eliminate intermediaries, ensuring better price realization for the farmer.

F. Crowd sourced Disease Mapping A community-driven module will be introduced to aggregate user diagnoses, creating real-time "disease heat maps" that alert farmers to potential outbreaks in their vicinity, enabling proactive pest management.

VIII. CONCLUSION

The proposed AgriAid ecosystem successfully bridges the critical gap between advanced agricultural intelligence and the practical, accessible needs of rural farming communities. By synthesizing high-precision Convolutional Neural Networks (CNNs) for disease detection [1] with the contextual reasoning capabilities of Generative AI, the system delivers a holistic advisory platform that surpasses the limitations of traditional unimodal solutions [6]. The implementation of a multimodal interface supporting voice, text, and image directly addresses the literacy and linguistic barriers that have historically hindered the adoption of digital agriculture tools. Furthermore, the system's offline-first architecture ensures reliable decision support in low-connectivity environments, resolving a persistent challenge identified in prior studies [7]. Ultimately, AgriAid democratizes access to agronomic expertise, empowering farmers with real-time, data-driven insights to enhance crop productivity and economic resilience.

REFERENCES

1. S.P.Mohanty,D.P.Hughes, and M.Salathé,“Using deep learning for image-based plant disease detection,” Frontiers in Plant Science, vol. 7, p. 1419, Sep. 2016. Available: https://www.frontiersin.org/articles/10.3389/fpls.2016.014
2. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, “Deep neural networks based recognition of plant diseases,” Computational Intelligence and Neuroscience, vol. 2016, Article ID 3289801, 2016. Available: https://doi.org/10.1155/2016/3289801
3. K.P.Ferentinos, “Deep learning models for plant disease detection and diagnosis,” Computers and Electronics in Agriculture, vol. 145, pp. 311–318, Feb. 2018. Available: https://doi.org/10.1016/j.compag.2018.01.009
4. J.G.A. Barbedo, “Factors influencing the use of image processinginplantdiseasediagnosis,”Computers and Electronics in Agriculture, vol. 154, pp. 296–302, Oct. 2018. Available: https://doi.org/10.1016/j.compag.2018.09.020
5. M.Brahimi,K.Boukhalfa,andA.Moussaoui,“ Deep learning for plant diseases: Detection and diagnosis,” Multimedia Tools and Applications, vol. 77, no. 8, pp. 9909–9929, Apr. 2018. https://www.google.com/searchq=https://doi.org/10.1007/s11042-017-0993-3
6. R. Patel, N. Joshi, and A. Mehta, “AgriBot: A conversational AI system for real-time crop monitoring and farmer assistance,” inProc. IEEE Int. Conf. on AI in Agriculture, 2021.Available: Google Scholar Search for Title
7. L. Chen, M. Gupta, and S.R. Prakash, “Voice-driven assistive technologies for agricultural decision-making in low-connectivity regions,” in Proc. Int. Conf. on Smart Agriculture and AI, 2022. Available: Google Scholar Search for Title
8. S. Kanakala and S. Ningappa, “Plant disease detection using hybrid CNN-LSTM architecture,”in Proc.Int.Conf.on ComputationalVisionandBio-InspiredComputing,2021. Available: GoogleScholar Search forTitle
9. A.S. Abade, P.A. Ferreira, and F.B. Vidal, “A comprehensive review of convolutional neural networks in plant disease identification,”Expert Systems with Applications, vol. 168, p. 114221, 2021. https://doi.org/10.1016/j.eswa.2020.114221
10. P.S. Thakur, P. Khanna, T. Sheorey, and A. Ojha, “PlantXViT: An interpretable plant disease classification model using vision transformers,” in Proc. IEEE Conf. on Smart Farming Technologies, 2023. Available:https://arxiv.org/abs/2207.07919