Problem Statement

Despite advancements in artificial intelligence, over 100 million Indian farmers still lack reliable, personalized crop guidance. Most existing tools ignore regional languages, crop-specific needs, and local soil conditions. They also overlook experienced farmers whose practical knowledge is not digitized. Current AI models often produce generic or incorrect responses, leading to crop damage and growing distrust in tech. Meanwhile, tools like ChatGPT help engineering students daily — but there's no equivalent for agriculture students or farmers. No system today learns from real farmers or adapts to local data. What's needed is a GenAI-powered assistant that uses farmer feedback, real-world context, and local inputs to deliver accurate, trustworthy advice.

Target Audience & Context

AgroGPT is built for small to mid-scale Indian farmers in rural and semi-urban regions who struggle with limited expert guidance and language-access barriers. It supports voice, text, and image input to ensure usability, even for non-tech-savvy users. By routing low-confidence queries to trusted peer farmers, AgroGPT builds a feedback loop grounded in community knowledge making AI more inclusive, local, and trusted.

Use of Gen-AI

AgroGPT harnesses Generative AI as both a responder and a learner. When a farmer submits a query (text, voice, or image), GPT-4 processes it and answers directly only if confident. If not, the query is routed to peer farmers via Firebase. Their responses are collected and ranked using semantic similarity, and GPT-4 summarizes and validates them into one reliable response. This human-powered data loop transforms every unanswered question into a community-labeled training instance. Over time, AgroGPT builds its own crop-specific, region-aware dataset of farmer-validated interactions, enabling future fine-tuning of a domain-specific LLM. The system also integrates real-time soil sensor data into the LLM's context, making answers grounded in field conditions. Unlike traditional AI tools that rely on static internet data, AgroGPT continuously improves by learning from real farmers, real soil, and real outcomes. This is GenAI that grows with the farmer.

Solution Framework

AgroGPT is a GenAI-powered assistant built for rural deployment, enabling farmers to query crop-related problems using a PWA that supports text (via input field), voice (via Whisper API), and image (via Google Vision API). The app runs on a React + Tailwind frontend, with Firebase Authentication and Firestore as its real-time database and storage. When a query is submitted, it is first analyzed using OpenAI's GPT-4 via API. A custom confidence threshold (e.g., based on token-level logprobs or GPT-4's function calling metadata) determines whether the model can reply directly. If confidence is low, the query is routed to peer farmers using Firebase Cloud Messaging (FCM). Responses from these farmers are collected asynchronously, filtered using Weaviate vector similarity, and optionally passed through a human moderator UI (moderator dashboard in Firebase Admin SDK). The filtered responses are summarized using GPT-4 with prompt chaining, which generates a final answer. That answer is returned to the farmer via both text and Google TTS voice output. An ESP32 board connected to a capacitive soil moisture sensor (and optional DHT11 for temperature/pH) logs real-time field data to Firebase using HTTP/MQTT. This context is injected into GPT-4's system prompt for soil-aware answers. All interactions are stored as structured documents in Firestore for dataset creation.

A points-based reward system (Gamify.js logic) ensures peer participation. This feedback loop continuously strengthens the model, making AgroGPT a truly farmer-trained AI.

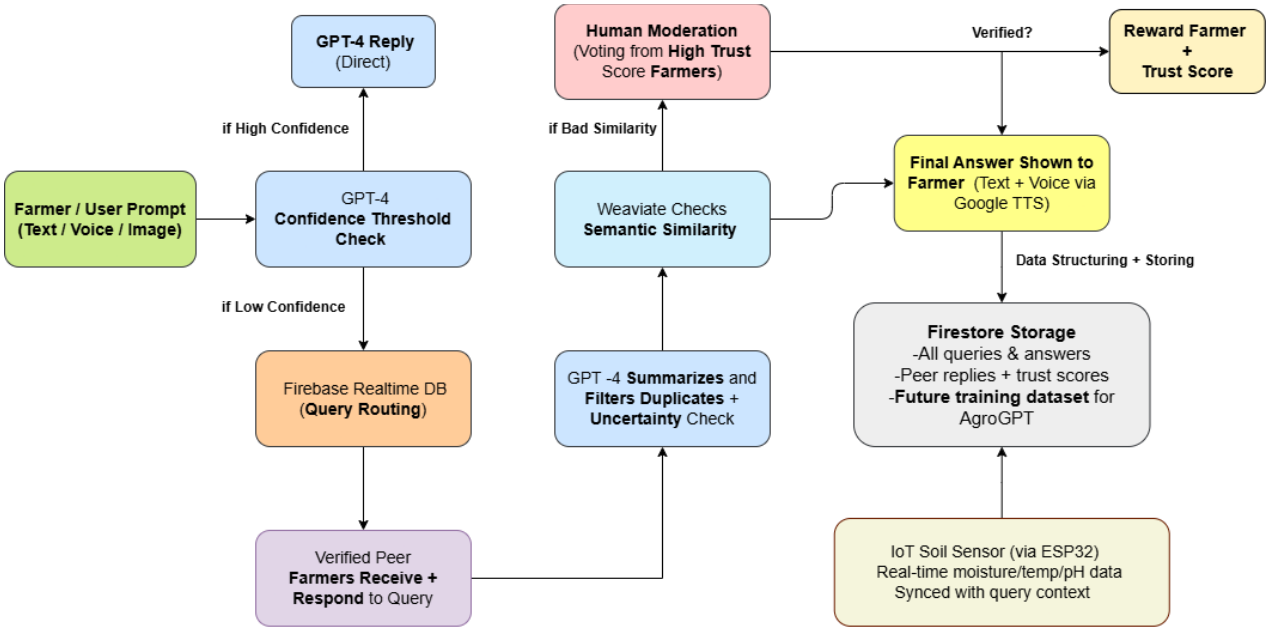

Figure 1: AgroGPT Architecture FLow

Feasibility & Execution

AgroGPT's core MVP can be built within 48 hours using production-ready tools. The GenAI pipeline runs via OpenAI's GPT-4 API and Google TTS. Firebase handles query routing and peer reply storage, while Firestore logs interaction history. A basic Weaviate similarity check or embedding match (using OpenAI embeddings) ranks peer responses. The PWA frontend, built with React and Tailwind, works on mobile and desktop. IoT data from a soil sensor (ESP32) can be simulated or hardcoded during the hackathon. The fallback-answer loop and GenAI summarization are fully API-driven, making it easy to demo the real-world potential, even with minimal infrastructure.

Scalability & Impact

AgroGPT is designed to benefit a wide range of users, from small farmers and agriculture students to agri-extension officers, NGOs, startups, and government advisory programs. Its language-agnostic GenAI, sensor integration, and peer validation make it highly adaptable to diverse geographies and crops. As usage grows, the platform builds a rich, region-specific dataset that improves its accuracy and trustworthiness. AgroGPT can serve as a plug-and-play advisory layer for public and private agri-initiatives, creating an ecosystem where even the most underserved voices shape the future of AI in farming.

Conclusion & MLP

AgroGPT is a Minimum Lovable Product because it delivers exactly what the end user needs: region-specific crop advice, voice-based interaction, sensor-driven context, and trusted answers from real farmers. It works offline, supports local languages, and rewards contributors — all with minimal infrastructure. Even at MVP stage, it solves a real, unmet need. With a little investment, AgroGPT can evolve into the most intelligent LLM in agriculture — trained not on scraped text, but on the lived experience of farmers themselves.