Here is a comprehensive Product Requirement Document (PRD) for the Automated Crop Insurance Platform.

**Product Requirement Document: Automated Crop Insurance Platform**

* **Version:** 1.0
* **Date:** June 18, 2025
* **Author:** Gemini
* **Status:** Draft
* **Stakeholders:** Head of Product, Lead Engineer, Insurer Partner Lead, Farmer Outreach Coordinator, Legal & Compliance

**1. Introduction**

**1.1. Problem Statement**

Traditional crop insurance is critical for farmers, yet the process is plagued by inefficiency. Claim processing is manual, slow (often taking months), and requires extensive paperwork and costly physical inspections. This leads to high administrative costs for insurers, long, uncertain waits for farmers, and a general lack of trust in the system. The ambiguity in data and process also creates opportunities for fraud, further increasing costs and premiums.

**1.2. Proposed Solution**

The Automated Crop Insurance Platform is a revolutionary system designed to provide instant, transparent, and data-driven insurance settlements. By leveraging a powerful combination of modern technologies, the platform will:

* Use **Blockchain** for immutable and instant verification of land ownership.
* Use **IoT and Satellite data** for objective, real-time weather condition monitoring (specifically rainfall).
* Use a customer-facing **AI Agent** to simplify and automate the claim filing process.
* Use a **Large Language Model (LLM) with Retrieval-Augmented Generation (RAG)** to ensure claims are instantly and accurately validated against policy conditions.

This technology stack removes ambiguity, eliminates manual verification, and builds a foundation of trust between the farmer and the insurer.

**1.3. Vision**

To create the world's most efficient and trusted agricultural insurance ecosystem, empowering farmers with financial security through transparent technology.

**2. Goals and Success Metrics**

**2.1. Product & Business Goals**

|  |  |
| --- | --- |
| **Goal Type** | **Goal Statement** |
| **Product** | Radically reduce claim processing time from months to minutes. |
| **Product** | Significantly improve farmer satisfaction and trust in the insurance process. |
| **Business** | Decrease operational costs associated with claim verification and management. |
| **Business** | Minimize fraudulent claims and improve the insurer's loss ratio. |
| **Business** | Establish a new standard for technology-led insurance products and capture significant market share. |

**2.2. Objectives and Key Results (OKRs)**

|  |  |
| --- | --- |
| **Objective** | **Key Result** |
| **1. Dramatically Accelerate Claim Settlement** | KR 1.1: Reduce average end-to-end claim settlement time from 90 days to under 24 hours. &lt;br> KR 1.2: Achieve 95% of all drought-related claims processed and decided without any human intervention. |
| **2. Enhance Farmer Trust and Adoption** | KR 2.1: Achieve a farmer Net Promoter Score (NPS) of +50 within the first year. &lt;br> KR 2.2: Successfully onboard and provide policies for 10,000 farmers in the pilot region within 12 months. |
| **3. Drive Unprecedented Operational Efficiency** | KR 3.1: Reduce the administrative cost per claim by 60% compared to the traditional paper-based process. &lt;br> KR 3.2: Reduce fraudulent payouts for drought-related claims to near-zero. |

**3. User Personas**

**3.1. Priya, the Farmer (Primary Persona)**

* **Demographics:** 35 years old, owns a 5-hectare farm in a rain-fed district of Maharashtra. Uses a smartphone for communication and payments.
* **Goals:**
  + Get quick, fair compensation when her crops fail due to drought.
  + Have a simple, easy-to-understand process without complex paperwork.
  + Trust that the data used for her claim (e.g., rainfall) is accurate and impartial.
* **Frustrations:**
  + "Last time, it took 6 months to get my claim money, and I had to visit the government office five times."
  + "I never know if my claim will be approved or why it was rejected."

**3.2. Anand, the Insurance Manager (Secondary Persona)**

* **Role:** Portfolio Manager at a partner insurance company.
* **Goals:**
  + Get an accurate, real-time overview of risk and liability across his portfolio.
  + Reduce the high operational costs of sending agents for farm inspections.
  + Minimize financial losses from fraudulent or inaccurate claims.
* **Frustrations:**
  + "Our biggest challenge is unpredictable, large-scale payouts after a bad season. We lack predictive insight."
  + "The cost of verifying thousands of small claims manually is enormous."

**4. Features & User Stories**

**Epic 1: Farmer Onboarding & Policy Management**

* **F-1.1:** As Priya, I want to create an account using my mobile number and a simple OTP so I can get started quickly.
* **F-1.2:** As Priya, I want to securely link my official digital identity (e.g., Aadhaar) to my account so the system knows who I am.
* **F-1.3:** As Priya, I want the platform to automatically fetch my land ownership details from the government's land registry blockchain so I don't need to upload any documents.
* **F-1.4:** As Priya, I want to view available insurance policies presented in a simple format, with a clear explanation of the drought threshold (e.g., "Coverage if rainfall is less than 100mm"), so I can understand what I'm buying.
* **F-1.5:** As Priya, I want to pay my insurance premium using common digital payment methods (like UPI) directly within the app.

**Epic 2: AI-Powered Claim Filing & Processing**

* **F-2.1:** As Priya, I want to start a claim by having a simple conversation with an AI Agent in my native language (e.g., Marathi) so the process feels easy.
* **F-2.2 (System):** The AI Agent must authenticate the user and automatically retrieve their active policy details upon claim initiation.
* **F-2.3 (System):** The AI Agent must perform an API call to the Blockchain Module to get a "Verified" or "Not Verified" status for land ownership.
* **F-2.4 (System):** The AI Agent must perform an API call to the Data Ingestion Module to retrieve the certified, aggregated rainfall data for the farmer's land coordinates during the policy period.
* **F-2.5 (System):** The AI Agent must query the Compliance Engine (LLM with RAG) with all verified data to receive a final APPROVED or REJECTED decision along with a detailed, clause-based explanation.
* **F-2.6:** As Priya, I want to receive an instant notification (via SMS and in-app) with the final claim decision and a simple, clear reason so I am not left wondering.
* **F-2.7:** As Priya, I want to see the status of my approved payment right in the app.

**Epic 3: Insurer Dashboard & Risk Analytics**

* **F-3.1:** As Anand, I want a real-time dashboard showing key metrics: total policies issued, active claims, automated approval/rejection rates, and total potential liability.
* **F-3.2:** As Anand, I want to be able to drill down into any individual claim and see its complete, immutable audit trail: the blockchain verification transaction, the exact weather data used, and the full explanation generated by the LLM.
* **F-3.3:** As Anand, I want to view the output from the predictive AI model, which uses weather forecasts to estimate the number and cost of claims for the upcoming season, so I can manage capital reserves.
* **F-3.4:** As Anand, I want to set up and receive automated alerts for anomalies, such as a sudden spike in claims from a non-drought-affected area.

**5. Non-Functional Requirements**

|  |  |
| --- | --- |
| **Category** | **Requirement** |
| **Performance** | The AI agent must respond to user queries in under 2 seconds. End-to-end automated claim processing must be completed in under 5 minutes. |
| **Scalability** | The architecture must support 100,000 active policies and handle up to 1,000 simultaneous claim submissions during peak events. |
| **Security** | All Personally Identifiable Information (PII) must be encrypted at rest and in transit. The platform must adhere to OWASP Top 10 security standards. |
| **Data Integrity** | Weather data sources must be certified and tamper-proof. All verification steps must be logged immutably. |
| **Reliability** | The system must have an uptime of 99.9%. |
| **Usability** | The farmer-facing interface must be intuitive, require minimal technical knowledge, and adhere to WCAG 2.1 accessibility standards. |
| **Localization** | The farmer UI and AI Agent must support English, Hindi, and Marathi for the pilot launch. |

**6. Out of Scope (For Version 1.0)**

To ensure a focused and successful launch, the following features will **not** be included in the initial release:

* **Insurance coverage for other perils:** The V1.0 product will exclusively handle claims for **drought (lack of rain)**. Floods, pests, fire, etc., are out of scope.
* **Manual claim submission:** There will be no paper-based or manual entry process. All claims must go through the AI agent.
* **Hardware provisioning:** The platform will not sell or manage physical IoT devices. It will only consume data from pre-approved, third-party sources.
* **Yield Guarantees:** The policy is weather-parametric (based on rainfall), not based on an assessment of actual crop yield.

**7. Assumptions and Dependencies**

* **Assumption:** Farmers in the target region have access to a smartphone and a basic data connection.
* **Dependency:** The existence of and ability to integrate with a government-managed (or officially recognized) digital Land Registry on a blockchain.
* **Dependency:** Consistent and reliable API access to certified meteorological data (e.g., from IMD, Skymet) for the target geographical areas.
* **Dependency:** A partnership with a licensed insurance underwriter who contractually agrees to honor the automated decisions made by the platform.

**8. Sign-off**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
|  | Head of Product |  |  |
|  | Lead Engineer |  |  |
|  | Insurer Partner Lead |  |  |
|  | Legal & Compliance |  |  |