

Aplicación multiplataforma de Gestión Agrícola y Análisis de Productividad para Granjas Antioqueñas.

**David Santiago Sierra Cadavid**

**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO**

**FACULTAD DE INGENIERÍA**

**TECNOLOGÍA DESARROLLO DE SOFTWARE**

**MEDELLÍN**

**2025**

Aplicación multiplataforma de Gestión Agrícola y Análisis de Productividad para Granjas Antioqueñas.

**Trabajo de grado para optar al título de  
TECNÓLOGO EN DESARROLLO DE SOFTWARE**

**Asesora:**

**LILIANA MARIA GARCIA AGUIRRE**

**Magister en Ingeniería de Software**

**David Santiago Sierra Cadavid**

**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO**

**FACULTAD DE INGENIERÍA**

**TECNOLOGÍA DESARROLLO DE SOFTWARE**

**MEDELLÍN**

**2025**

# Contenido

Pág.

<b>Lista de figuras</b> .....	5
<b>Lista de tablas</b> .....	6
<b>Resumen</b> .....	7
<b>Abstract</b> .....	8
<b>Glosario</b> .....	9
<b>Introducción</b> .....	10
<b>1. Planteamiento del problema</b> .....	12
<b>1.1 Descripción</b> .....	12
<b>1.2 Formulación</b> .....	13
<b>2. Justificación</b> .....	14
<b>3. Objetivos</b> .....	15
<b>3.1 Objetivo general</b> .....	15
<b>3.2 Objetivos específicos</b> .....	15
<b>4. Marco teórico</b> .....	16
<b>4.1. Antecedentes</b> .....	16
<b>4.2 Aplicaciones agrícolas existentes</b> .....	17
<b>4.3 Otras aplicaciones</b> .....	18
<b>4.4 Análisis comparativo de proyectos existentes</b> .....	19
<b>5. Metodología</b> .....	20
<b>5.1 Tipo de proyecto</b> .....	20
<b>5.2 Método</b> .....	20
<b>5.2.1 Instrumentos de recolección de información</b> .....	21
<b>5.2.2 Fuentes primarias.</b> .....	21
<b>5.2.3 Fuentes secundarias.</b> .....	22
<b>6. Resultados</b> .....	23
<b>6.1 Planificación - Scrum</b> .....	23
<b>6.1.2 Descripción de las Historias de Usuario</b> .....	28
<b>6.2 Arquitectura y Diseño</b> .....	34
<b>6.2.2 Arquitectura MVVM</b> .....	34

<b>6.2.3 Diseño de Base de datos</b> .....	37
<b>6.3 Implementación</b> .....	37
<b>6.4 Pruebas</b> .....	39
<b>6.3.5 Casos de pruebas</b> .....	41
<b>6. Conclusiones</b> .....	46
<b>7. Recomendaciones</b> .....	47
<b>8. Referencias Bibliográficas</b> .....	48

## Lista de figuras

Pág.

Figura 1. Etapas SCRUM .....	21
Figura 2. MVVM en flutter.....	36
Figura 3. Diagrama entidad relación granjas .....	37
Figura 4. Programación registro usuario.....	39
Figura 5. CRUD granjas .....	39
Figura 6. Pruebas unitarias.....	40

## Lista de tablas

Tabla 1. Spring1_Gestión de usuarios .....	24
Tabla 2. Sprint 2 Gestión de Lotes.....	25
Tabla 3. Spring 3. Registro de Actividades .....	26
Tabla 4. Historio de Usuario 1 .....	28
Tabla 5. Historia de Usuario 2 .....	28
Tabla 6. Historia de Usuario 3 .....	29
Tabla 7. Historia de Usuario 4 .....	29
Tabla 8. Historia de Usuario 5 .....	30
Tabla 9. Historia de usuario 6 .....	31
Tabla 10. Historia de Usuario 7 .....	31
Tabla 11. Historia de Usuario 8 .....	32
Tabla 12. Historia de Usuario 9 .....	32
Tabla 13. Historia de Usuario 10 .....	33
Tabla 14. Historia de Usuario 11 .....	34
Tabla 15. Caso de prueba_InicioSesión.....	41
Tabla 16. Caso prueba_CRUD Granja.....	42
Tabla 17. Caso de prueba_ Creación lote .....	43

## Resumen

Este proyecto de software fue desarrollado para mejorar el seguimiento y la gestión de granjas agrícolas, permitiendo registrar, controlar y visualizar información clave sobre los procesos productivos. El sistema utiliza como base una estructura de datos que incluye tablas para lotes, cultivos, insumos, actividades y registros de producción, lo que facilita el manejo organizado y actualizado de la información agrícola.

El desarrollo se realizó empleando la metodología ágil SCRUM, organizando el trabajo en sprint cortos que permitieron entregar funciones incrementales, validar constantemente con el usuario y adaptarse a las necesidades reales de la granja. Durante cada iteración se realizaron reuniones de planificación, revisión y retrospectiva para garantizar un avance continuo y mejorar la calidad del producto.

La aplicación fue programada en Flutter, lo que permitió crear una interfaz moderna, ágil y multiplataforma, compatible con dispositivos móviles y tablets utilizados en campo. Para el almacenamiento de la información se utilizó Firebase Firestore, una base de datos en la nube que ofrece sincronización en tiempo real, alta disponibilidad y facilidad de integración con Flutter.

El resultado es una herramienta que permite registrar actividades agrícolas, controlar cultivos, llevar historial de labores, monitorear el progreso de los lotes y generar información útil para la toma de decisiones. Gracias a su arquitectura flexible y su enfoque ágil, el software se adapta a diferentes tipos de granjas y contribuye a mejorar la eficiencia y trazabilidad del proceso productivo.

## **Abstract**

This software project was developed to improve the monitoring and management of agricultural farms, allowing users to record, control, and visualize key information about production processes. The system is based on a data structure that includes tables for lots, crops, supplies, activities, and production records, which facilitates organized and up-to-date management of agricultural information.

The development was carried out using the SCRUM agile methodology, organizing the work into short sprints that enabled the delivery of incremental features, constant validation with the user, and adaptation to the real needs of the farm. During each iteration, planning, review, and retrospective meetings were held to ensure continuous progress and improve product quality.

The application was programmed in Flutter, allowing the creation of a modern, agile, and multiplatform interface, compatible with mobile devices and tablets used in the field. For data storage, Firebase Firestore was used, a cloud database that provides real-time synchronization, high availability, and easy integration with Flutter.

The result is a tool that allows users to record agricultural activities, monitor crops, keep a history of tasks, track the progress of lots, and generate useful information for decision-making. Thanks to its flexible architecture and agile approach, the software adapts to different types of farms and helps improve the efficiency and traceability of the production process.

## Glosario

**Firestore:** Plataforma de desarrollo de aplicaciones de Google que ofrece servicios en la nube como bases de datos en tiempo real, autenticación, almacenamiento, hosting y analítica, permitiendo crear apps rápidas sin necesidad de gestionar servidores.

**Flutter:** Framework de código abierto creado por Google para desarrollar aplicaciones móviles, web y de escritorio desde una sola base de código, usando el lenguaje Dart y un sistema propio de widgets rápidos y personalizables.

**Granja agrícola:** Unidad de producción donde se cultivan plantas o se crían animales con fines alimentarios, comerciales o de subsistencia, utilizando recursos como tierra, agua y herramientas agrícolas.

**MVVM (Model–View–ViewModel):** Patrón de arquitectura que separa la interfaz, la lógica de presentación y los datos en tres capas: **Model**, **View** y **ViewModel**, permitiendo aplicaciones más organizadas, mantenibles y fáciles de probar.

**Proceso de producción agrícola:** Conjunto de actividades mediante las cuales se preparan, siembran, cultivan y cosechan productos del campo, utilizando recursos como tierra, agua, insumos y técnicas agrícolas para obtener alimentos u otros bienes vegetales.

## Introducción

El presente proyecto se enfoca en el desarrollo de un sistema de software para la gestión y seguimiento integral de granjas agrícolas, con el propósito de mejorar la organización, control y análisis de los procesos productivos, así como optimizar la toma de decisiones basada en información precisa y actualizada. Este trabajo se realiza en un contexto en el que la digitalización del sector agropecuario se ha vuelto fundamental para incrementar la eficiencia, la productividad y la competitividad, respondiendo a los desafíos que enfrentan los productores rurales frente a la mecanización limitada, la escasa disponibilidad de información y la necesidad de sostenibilidad. La Agricultura 4.0, también conocida como agricultura digital o inteligente, proporciona el marco conceptual de este proyecto, integrando tecnologías emergentes como Big Data, análisis predictivo, aplicaciones móviles y blockchain para mejorar la trazabilidad de los datos, la automatización de procesos y la eficiencia de la gestión agrícola (Ovalle Másmela, Romero-Perdomo, & Uribe Galvis, 2023).

Durante el desarrollo del proyecto, se elaboró un marco teórico, en el cual se analizaron y validaron antecedentes de proyectos y aplicaciones agrícolas existentes, así como conceptos relacionados con la agricultura de precisión, la gestión de cultivos y la digitalización del campo. Este análisis permitió identificar buenas prácticas, limitaciones de soluciones previas como Agroptima, Isagri y AgroTIC, y elementos clave para el diseño de un sistema que integre gestión de inventarios, control de actividades productivas y generación de reportes automáticos, adaptados a las necesidades específicas de las granjas regionales.

El proyecto se ejecutó bajo la metodología ágil SCRUM, organizando el trabajo en sprints iterativos que facilitaron la entrega incremental de funcionalidades, la retroalimentación continua de los usuarios y la adaptación del sistema a requerimientos cambiantes.

La aplicación se desarrolló utilizando Flutter, lo que permitió crear una interfaz multiplataforma, moderna y accesible, y se implementó la base de datos en Firebase Firestore, garantizando sincronización en tiempo real, disponibilidad en la nube y seguridad de la información.

Entre las limitaciones del proyecto se encuentran la dependencia de conexión a internet para el

funcionamiento óptimo de la base de datos en la nube, el alcance inicial centrado en determinadas actividades agrícolas, y la ausencia de módulos avanzados de análisis predictivo o automatización de decisiones, los cuales podrán incorporarse en futuras versiones.

Este proyecto busca fortalecer la digitalización del sector agrícola regional, ofreciendo herramientas que promuevan la eficiencia operativa, la toma de decisiones basada en datos y la sostenibilidad, contribuyendo a mejorar la productividad y la calidad de vida de los agricultores.

# 1. Planteamiento del problema

## 1.1 Descripción

La necesidad de modernizar la gestión y el monitoreo de las actividades agrícolas en Antioquia surge de los desafíos que enfrentan los agricultores al utilizar métodos tradicionales y la oportunidad de mejorar la eficiencia, la productividad y la toma de decisiones mediante la adopción de tecnologías digitales. En la agricultura, históricamente, se han aplicado metodologías de cultivo de manera tradicional debido a la falta de fácil acceso a la información, a las técnicas de innovación actuales y a los avances tecnológicos en predicción, precisión o apoyo agrícola. Esto puede llevar a decisiones equivocadas sobre qué cultivos sembrar, generando pérdidas de producto e inversión

Las granjas antioqueñas requieren de un monitoreo constante para asegurar una producción óptima. La gestión agrícola en la región sigue dependiendo de métodos manuales y registros dispersos, lo que dificulta la recolección y el análisis de información clave. Esta falta de acceso a datos en tiempo real impide la identificación temprana de problemas y limita la capacidad de los agricultores para tomar decisiones informadas.

Actualmente, la gestión se realiza con herramientas poco eficientes, como registros físicos o sistemas no integrados, lo que genera errores, pérdida de información y dificultades en el análisis de datos.

Mejorar estos procesos es clave para optimizar la toma de decisiones, reducir pérdidas y aumentar la productividad de las granjas. Un acceso más estructurado a la información permitiría a los agricultores monitorear sus cultivos y recursos en tiempo real, anticiparse a problemas y mejorar la eficiencia en la gestión de insumos. Además, el análisis de datos ayudaría a identificar patrones y tendencias que contribuyan a la mejora continua de la producción.

## **1.2 Formulación**

¿Cómo el desarrollo de una aplicación multiplataforma puede apoyar la gestión integral, el monitoreo y el aumento de la productividad en una granja antioqueña?

## 2. Justificación

Las granjas son unidades productivas agrícolas que consisten en terrenos dedicados a la siembra, cultivo de productos agrícolas, o la cría de animales. En ellas, los agricultores realizan actividades como la siembra, cosecha, riego, y control de plagas, entre otras tareas, jugando un papel crucial en la economía y el bienestar de la sociedad. Aportan alimentos frescos y productos de alta calidad, generan empleo directo e indirecto, y son una fuente importante de ingresos para las familias rurales.

El sector agropecuario constituye uno de los pilares fundamentales para el desarrollo económico y social del país, dado que provee bienes e insumos esenciales para las demás actividades productivas. Su fortalecimiento es clave para garantizar la seguridad alimentaria, la generación de empleo rural y la sostenibilidad de los recursos naturales. En este sentido, el Banco de la República (2015, citado en Colombia Aprende, 2021) señala que *“el sector primario proporciona bienes e insumos a los demás sectores de la economía, por lo que se constituye en un eje fundamental para el desarrollo y progreso de un país”* (p. 17). Esta afirmación evidencia la relevancia de impulsar iniciativas tecnológicas que mejoren la gestión y productividad del sector, promoviendo la competitividad y la innovación en las prácticas agropecuarias.

En este contexto, la implementación de una herramienta tecnológica para el registro y monitoreo de las actividades productivas se convierte en una necesidad urgente. La herramienta permitirá a los usuarios registrar detalladamente sus actividades, desde la siembra hasta la cosecha, pasando por la gestión de recursos, fertilizantes y plaguicidas. El análisis de la información registrada permitirá programar de manera precisa los tiempos de siembra, riego y fertilización, lo que contribuirá a reducir el desperdicio de agua, semillas y fertilizantes. Además, facilitará la toma de decisiones basadas en la experiencia y los datos históricos, optimizando los recursos, disminuyendo los costos operativos y mejorando la calidad de los productos, entre otros beneficios.

Una de las características más relevantes que quieren en esta herramienta es que sea fácil de usar. Ha sido diseñada para ser intuitiva y accesible, de modo que tanto las personas con poca experiencia tecnológica, como los técnicos especializados, puedan aprovechar sus funcionalidades sin dificultad. De esta manera, se busca garantizar la inclusión de los campesinos con menor familiaridad con la tecnología y, al mismo tiempo, brindar eficiencia a los usuarios con mayor nivel técnico.

## **3. Objetivos**

### **3.1 Objetivo general**

Desarrollar una aplicación multiplataforma que mejore la gestión y el monitoreo de las actividades productivas en granjas del departamento de Antioquia, mediante el registro de datos en tiempo real, el control de inventarios y la generación de reportes, buscando el aumento de la eficiencia operativa.

### **3.2 Objetivos específicos**

- Identificar las necesidades que tienen los agricultores para sistematizar los procesos de producción de sus granjas agrícolas.
- Diseñar una interfaz de usuario intuitiva y accesible, lo que facilitará la adopción de la aplicación por parte de los usuarios.
- Implementar una base de datos que permita almacenar y analizar la información registrada por los usuarios.
- Desarrollar un módulo de análisis de datos para la identificación de patrones y tendencias, lo que contribuirá a la toma de decisiones informadas.
- Validar la funcionalidad y usabilidad de la aplicación a través de pruebas con usuarios reales.

## 4. Marco teórico

### 4.1. Antecedentes

La agricultura, pilar fundamental de la economía, enfrenta el desafío de la digitalización para optimizar sus procesos y aumentar la productividad. La Agricultura 4.0 surge como una respuesta a esta necesidad, integrando tecnologías como el análisis de datos. Las aplicaciones móviles se han convertido en herramientas clave para los agricultores, brindando acceso a información en tiempo real, facilitando la toma de decisiones y optimizando las operaciones agrícolas. (Cerezo, n.d.)

El documento "SISTEMA DE RECOMENDACIÓN INTELIGENTE PRE-SIEMBRA EN ZONAS RURALES DEL MUNICIPIO DE SOACHA Y SIBATÉ.pdf" tiene como objetivo principal desarrollar un prototipo de software de un sistema de recomendación pre-siembra para agricultores de los municipios de Soacha y Sibaté. Este prototipo se enfoca en tres cultivos específicos: papa, fresa y arveja, elegidos por su importancia estratégica en la región.

La finalidad del sistema es brindar una herramienta de apoyo a la agricultura de precisión (AP), facilitando la toma de decisiones de los agricultores al recomendar el cultivo más apto para la siembra y producción, considerando factores como la zona, el lugar, la oferta y la demanda de los productos. Se espera que esta herramienta mejore la calidad de vida de los agricultores.

El desarrollo del prototipo se abordó desde la Ingeniería de Software, cubriendo aspectos como el planteamiento del problema, la justificación, antecedentes de sistemas similares, el alcance y las conclusiones. Se define un sistema de recomendación como un software que ayuda a los usuarios a obtener el producto más adecuado a sus preferencias y necesidades.

La aplicación móvil desarrollada incluye funcionalidades como registro e inicio de sesión de usuarios (e incluso como invitado), un menú principal con acceso a recomendaciones, perfil de usuario, recomendaciones guardadas y la opción de cerrar sesión. El módulo de recomendaciones muestra información relevante para la pre-siembra del cultivo de interés, incluyendo las mejores fechas de siembra, el tiempo de cosecha y la demanda actual. La información de los usuarios y las recomendaciones se almacenan en una base de datos relacional SQLite. (Gonzales Camargo & Sepúlveda Ospina, 2022)

El documento, elaborado por Juan Ovalle Másmela, Felipe Romero-Perdomo y Claudia Uribe Galvis para la Corporación Colombiana de Investigación Agropecuaria

(AGROSAVIA), se centra en el análisis del uso potencial y el desarrollo de tecnologías emergentes en el sector agropecuario colombiano. El texto sitúa la discusión en el contexto de la evolución tecnológica de la agricultura, mencionando las cinco "revoluciones" industriales: la Mecanización 1.0 (agua y vapor), la Electrificación 2.0 (producción en masa con electricidad), la Automatización 3.0 (ingeniería electrónica, telecomunicaciones y computadoras), la Digitalización 4.0 (sistemas físicos cibernéticos, dispositivos conectados desde 2011) y la Personalización 5.0 (interdependencia hombre-máquina, computación cognitiva, inteligencia humana desde 2021). La Agricultura 4.0, también conocida como Agricultura Digital o Inteligente (*Digital Farming / Smart Farming*), lleva desarrollándose aproximadamente una década y se fundamenta en la digitalización de la gestión agrícola para mejorar la precisión de los procesos automatizados y la trazabilidad de datos.

Entre las tecnologías emergentes clave que se abordan en el documento se encuentran:

**Big Data:** Implica el análisis de grandes volúmenes de datos de diversas fuentes (sensores, informes gubernamentales, servicios web, encuestas, redes sociales) para descubrir patrones, correlaciones, tendencias y asistir en la toma de decisiones. Las aplicaciones incluyen pronósticos (rendimiento, plagas, enfermedades, clima), análisis de precios y rentabilidad, y modelos para políticas de sostenibilidad.

**Cadena de Bloques (Blockchain):** Un libro compartido que registra el histórico de acciones de forma segura y transparente. Permite acceso a datos en tiempo real sobre la cadena de suministro (calidad, clima, pagos, demanda, precio) mejorando la trazabilidad y la seguridad. (Ovalle Másmela, Romero-Perdomo, & Uribe Galvis, 2023)

La baja implementación de tecnología en el campo colombiano es preocupante, especialmente si se considera la rápida transición global hacia la transformación digital. Los esfuerzos gubernamentales pasados en el desarrollo agropecuario han sido catalogados como ineficientes, lo que ha generado una enorme brecha de desigualdad en el sector rural. Ante esta realidad, se reconoce que la innovación se erige como una vía crucial para abordar estas problemáticas complejas, y la capacidad de aplicar tecnologías digitales no solo mejora la resiliencia del agro frente a los retos climáticos y la desigualdad, sino que también impulsa la productividad y la competitividad en un mundo digitalmente transformado. La implementación de soluciones sistemáticas y sostenibles es imperativa para resolver la "bola de nieve" de problemáticas actuales. (José Leonardo Mejía Castillo, 2024)

## **4.2 Aplicaciones agrícolas existentes**

Diversas aplicaciones ya se encuentran disponibles en el mercado, ofreciendo una variedad de servicios. Algunas de ellas se centran en la gestión integral de la explotación, como

Agroptima e Isagri, permitiendo llevar un registro de las actividades, controlar gastos y generar informes de rentabilidad. Otras se enfocan en el análisis de datos e imágenes satelitales, como Auravant, para la implementación de la agricultura de precisión. Existen también aplicaciones para el diagnóstico de enfermedades y plagas, como Agrio, que utilizan la IA para identificar problemas a partir de fotografías y brindar recomendaciones de tratamiento.

### **4.3 Otras aplicaciones**

Agroptima: Es un software agrícola en la nube que permite la gestión agronómica y económica de la explotación agrícola, facilitando el cumplimiento de las leyes agrícolas. Permite introducir datos de la explotación, tratamientos fitosanitarios, aplicaciones de fertilizantes, rendimientos, llevar un cuaderno de campo digital, extraer informes para la administración pública o para análisis históricos (control de costes, análisis por parcela), y crear órdenes de trabajo. Utiliza una interfaz GIS que visualiza las parcelas sobre un mapa (Agroptima, n.d.)

Isagri: Tiene un fundamento similar a Agroptima, permitiendo la recolección de datos para generar un cuaderno de campo/explotación, certificación Global GAP, informes para la optimización de la explotación y gestión de costes. También dispone de un módulo para la gestión de explotaciones ganaderas. (*Isagri*, n.d.)

AgroTIC es un ejemplo relevante de una aplicación basada en teléfonos inteligentes diseñada para brindar asistencia técnica agrícola remota en Colombia. Esta aplicación comprende cuatro módulos principales que ilustran la integración de funcionalidades clave:

- El Módulo de Chat facilita la comunicación entre agricultores, así como entre agricultores y agrónomos para asistencia técnica remota, y entre agricultores y comerciantes para la negociación de productos. Esto promueve una red de comunicación que permite a los agricultores compartir experiencias y consultar sobre problemas comunes de cultivos, un elemento crucial para la colaboración y el intercambio de conocimientos.
- El Módulo de Diagnóstico permite a los agricultores tomar fotografías de las hojas de las plantas, utilizando algoritmos de índices de vegetación visible y una Red Neuronal Convolutiva (CNN) para estimar la salud general de la planta y diagnosticar enfermedades. Los resultados son enviados a un agrónomo para un análisis más profundo, lo que representa un avance significativo en la identificación temprana de problemas.
- El Módulo de Producción permite a los agricultores llevar un registro detallado de sus tierras y cultivos, cuya información se comparte con los módulos de Diagnóstico y Marketing. Esto es esencial para estimar volúmenes de producción y preparar la información para el proceso de comercialización (AgroTIC, n.d.)

#### **4.4 Análisis comparativo de proyectos existentes**

Las herramientas Agroptima, Isagri y AgroTIC representan soluciones tecnológicas orientadas a la digitalización del sector agropecuario, centradas principalmente en la gestión de datos productivos, el monitoreo de cultivos y la optimización de los recursos agrícolas.

Agroptima destaca por ser un software agrícola en la nube que integra tanto la gestión agronómica como la económica de las explotaciones. Su enfoque está en el cumplimiento normativo, la trazabilidad de insumos y la generación de informes administrativos, incorporando además una interfaz GIS para la visualización geográfica de parcelas. Esta herramienta resulta particularmente útil en el contexto de la agricultura tecnificada y de mediana o gran escala.

Por su parte, Isagri comparte características similares, orientándose hacia la recopilación y sistematización de información para la creación de cuadernos de campo, la certificación Global GAP y la gestión de costes. A diferencia de Agroptima, incorpora un módulo adicional para la administración de explotaciones ganaderas, ampliando su cobertura hacia otros subsectores del ámbito agropecuario. Sin embargo, ambos sistemas presentan limitaciones en cuanto a su accesibilidad para pequeños productores o comunidades rurales con baja alfabetización digital.

En el caso de AgroTIC, se observa un enfoque más inclusivo y contextualizado para el entorno colombiano. Esta aplicación móvil combina herramientas de comunicación, diagnóstico, producción y comercialización, integrando algoritmos de inteligencia artificial (como redes neuronales convolucionales) para la detección temprana de enfermedades en cultivos. Su mayor fortaleza radica en la asistencia técnica remota y la colaboración entre agricultores, aunque su funcionalidad se centra principalmente en la fase agrícola y no abarca de manera integral la gestión administrativa y de inventarios de las explotaciones.

Frente a estas soluciones, el proyecto propuesto busca desarrollar una aplicación multiplataforma que combine los elementos más efectivos de las anteriores, pero con un enfoque más integral y adaptado a las necesidades de las granjas antioqueñas. Su factor diferenciador radica en la gestión unificada de las actividades productivas, incorporando el registro de datos en tiempo real, el control de inventarios y la generación automatizada de reportes. Además, su diseño priorizará la facilidad de uso y la accesibilidad, garantizando que tanto campesinos con poca familiaridad tecnológica como técnicos especializados puedan utilizarla sin dificultad. De esta forma, se busca promover la eficiencia operativa, la toma de decisiones basadas en datos y la digitalización del sector agropecuario regional.

## **5. Metodología**

### **5.1 Tipo de proyecto**

La Investigación Aplicada se define como trabajos originales realizados para adquirir nuevos conocimientos, dirigidos fundamentalmente hacia un objetivo o propósito específico práctico. En este sentido, el desarrollo de una aplicación multiplataforma que mejore la gestión y el monitoreo de las actividades agrícolas en granjas antioqueñas, mediante el registro de datos en tiempo real, el control de inventarios y la generación de reportes, busca directamente solucionar problemas concretos en el sector productivo

### **5.2 Método**

La planeación y ejecución del proyecto de grado se realizará mediante la implementación de la metodología ágil SCRUM. Este enfoque define la estrategia a seguir y el procedimiento para dar respuesta al problema y a los objetivos planteados. SCRUM es ideal para proyectos de desarrollo de software, ya que permite flexibilidad y adaptación a los cambios, promueve entregas de valor tempranas y continuas, y mejora la comunicación y colaboración del equipo.

La planeación y ejecución del proyecto de grado se realizará mediante la implementación de la metodología ágil SCRUM. Este enfoque define la estrategia a seguir y el procedimiento para dar respuesta al problema y a los objetivos planteados. SCRUM es ideal para proyectos de desarrollo de software, ya que permite flexibilidad y adaptación a los cambios, promueve entregas de valor tempranas y continuas, y mejora la comunicación y colaboración del equipo.

La metodología se estructura en una serie de Sprints, donde el Sprint se conoce como un período de tiempo fijo, generalmente de una a cuatro semanas, durante el cual el

equipo de desarrollo trabaja para completar un conjunto de tareas o funcionalidades priorizadas del Product Backlog. Su objetivo es entregar un incremento de producto funcional y potencialmente entregable al finalizar el ciclo. Durante el Sprint, no se deben realizar cambios que afecten el objetivo definido, y el equipo se enfoca en cumplir el Sprint Goal (meta del sprint). Cada Sprint incluye actividades como la planificación (Sprint Planning), el seguimiento diario (Daily Scrum), la revisión del incremento (Sprint Review) y la retrospectiva (Sprint Retrospective), las cuales permiten mejorar el producto y el proceso de trabajo de manera continua, en la figura a continuación se pueden evidenciar las etapas.

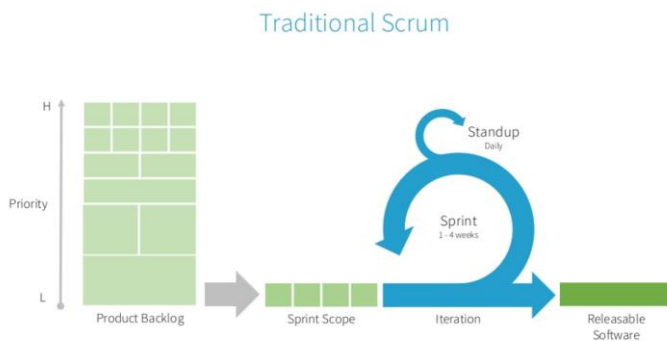


Figura 1. Etapas SCRUM

### 5.2.1 Instrumentos de recolección de información

### 5.2.2 Fuentes primarias.

Las fuentes primarias utilizadas en este proyecto incluyen la aplicación de encuestas a agricultores, la observación directa de los cultivos y el análisis de documentos especializados proporcionados por los propios productores. Estas actividades permitieron obtener información de primera mano sobre las necesidades reales del sector agrícola y los procesos que se llevan a cabo en el entorno productivo. Las observaciones y

recolección de datos se realizaron en diversas fincas ubicadas en el municipio de Girardota, lo que permitió comprender en contexto las dinámicas locales y las condiciones particulares del cultivo.

### **5.2.3 Fuentes secundarias.**

Las fuentes secundarias del proyecto están conformadas por informes técnicos del Ministerio de Agricultura, así como por artículos y documentos académicos de proyectos similares desarrollados en diversas universidades. También se consultó información proveniente de empresas del sector agrícola y se realizó una revisión de otras aplicaciones y herramientas digitales disponibles en el mercado, con el fin de identificar funcionalidades, tendencias y buenas prácticas tecnológicas. Toda esta información se encuentra ampliada y analizada en el marco teórico del documento.

## 6. Resultados

### 6.1 Planificación - Scrum

La planificación del proyecto se llevó a cabo siguiendo los principios de la metodología ágil SCRUM, lo que permitió organizar el trabajo en ciclos iterativos y entregar valor de manera continua. Cada Sprint fue definido como un período de tiempo, durante el cual el equipo se enfocó en completar un conjunto de tareas priorizadas del Product Backlog.

#### Product Backlog

El término Backlog (trabajo pendiente o pila de tareas) se refiere a aquellas tareas que requieren nuestra atención o al trabajo pendiente de un proyecto. En el contexto de la metodología ágil Scrum, el backlog es un artefacto crucial que ayuda al equipo de desarrollo a comprender cuáles son las tareas pendientes que deben completarse.

La metodología ágil Scrum estructura el desarrollo en ciclos cortos de trabajo llamados Sprints. En este marco, se distinguen dos tipos principales de backlog:

1. Product Backlog (Pila de Producto): Es el término genérico que recopila todo lo que se cree necesario para desarrollar el producto. En
2. Sprint Backlog (Trabajo Pendiente del Sprint): Es una lista de tareas seleccionadas que el equipo Scrum planea finalizar durante un sprint específico.

En este proyecto, la planificación y el seguimiento del trabajo se articulan mediante los backlogs:

- Estructura Ágil: El proyecto utiliza un cronograma que divide el desarrollo en Sprints numerados (como Sprint 1, Sprint 2, etc.), con una duración establecida.
- Product Backlog del Proyecto: La lista priorizada de funcionalidades y tareas requeridas se organiza en el Product Backlog.

### 6.1.1 Producto Backlog separado Spring con sus historias de usuario

Teniendo en cuenta que para el proyecto de la granja agrícola se está trabajando con la metodología SCRUM, a continuación, se presenta el Backlog del producto, organizado por cada Sprint junto con sus respectivas historias de usuario.

#### Sprint 1: Gestión de Usuarios y Granjas

**Objetivo:** Permitir el registro, inicio de sesión y gestión básica de granjas.

Tabla 1. Spring1\_Gestión de usuarios

Código	Historia de Usuario	Tareas Técnicas	Criterios de Aceptación
HU-01-S1	Como agricultor, quiero registrarme en la aplicación con mis datos, para poder acceder a la información de mi granja.	Crear formulario de registro en Flutter.- Validar datos de usuario.- Conectar a Firebase Authentication.	Registro exitoso con correo y contraseña.- Mensaje de error si faltan datos o correo ya existe.
HU-02-S1	Como agricultor, quiero iniciar y cerrar sesión en la aplicación, para proteger mi información personal.	- Implementar login con Firebase.- Implementar cierre de sesión.- Manejar estados de usuario logueado/no	- El usuario puede iniciar sesión y acceder a su información.- Cierre de sesión efectivo.

		logueado.	
HU-03-S1	Como agricultor, quiero agregar y editar la información de mi granja, para mantener un registro actualizado.	Crear CRUD de granjas.- Guardar datos en Firestore.- Validar campos obligatorios.	Se pueden crear, editar y eliminar granjas.- Datos persistentes en Firestore.
HU-04-S1	Como administrador, quiero asignar permisos a los usuarios, para controlar el acceso a la información.	- Definir roles en Firestore.- Implementar lógica de permisos en la app.	- Usuarios con permisos limitados no pueden acceder a funciones administrativas.

## Sprint 2: Gestión de Lotes y Cultivos

**Objetivo:** Registrar y controlar los lotes y cultivos de la granja.

Tabla 2. Sprint 2 Gestión de Lotes

<b>Código</b>	<b>Historia de Usuario</b>	<b>Tareas Técnicas</b>	<b>Criterios de Aceptación</b>
HU-01-S2	Como agricultor, quiero crear, editar y eliminar lotes de mi granja, para	- Crear CRUD de lotes.- Asociar lotes a la granja del usuario.- Guardar en Firestore.	- Los lotes se crean, editan y eliminan correctamente.- Cada lote está vinculado a la granja correcta.

	organizar la información del terreno.		
HU-02-S2	Como agricultor, quiero registrar cultivos asociados a cada lote, para tener un seguimiento detallado de la producción.	- Crear formulario de cultivos.- Asociar cultivos a lotes.- Guardar fechas y tipos de cultivo en Firestore.	- Los cultivos se asocian al lote correcto.- Información visible en detalle de lote.
HU-03-S2	Como agricultor, quiero consultar el estado y progreso de cada cultivo, para planificar actividades y cosechas.	- Implementar vista de progreso.- Mostrar fechas de siembra y estimación de cosecha.- Actualizar estado automáticamente.	- Estado y fechas se muestran correctamente.- Información actualizada en tiempo real.

### **Sprint 3: Registro de Actividades y Control de Inventarios**

**Objetivo:** Monitorear actividades agrícolas e inventarios de insumos.

*Tabla 3. Spring 3. Registro de Actividades*

<b>Código</b>	<b>Historia de</b>	<b>Tareas</b>	<b>Criterios de</b>
---------------	--------------------	---------------	---------------------

	<b>Usuario</b>	<b>Técnicas</b>	<b>Aceptación</b>
HU-01-S3	Como agricultor, quiero registrar actividades agrícolas por lote (fertilización, riego, cosecha), para llevar un historial completo.	- Crear formulario de actividad.- Guardar en Firestore con fecha y lote.- Mostrar historial por lote.	- Actividades registradas aparecen en historial.- Datos completos y ordenados por fecha.
HU-02-S3	Como agricultor, quiero registrar y actualizar inventarios de insumos (fertilizantes, semillas, herramientas), para controlar los recursos.	- Crear CRUD de inventarios.- Actualizar cantidades al usar insumos.- Guardar en Firestore.	- Inventario reflejado en tiempo real.- Alertas si stock bajo.
HU-03-S3	Como agricultor, quiero generar reportes de producción y consumo de insumos, para analizar eficiencia y costos.	- Implementar módulo de reportes.- Filtrar por fechas y lotes.- Mostrar gráficos simples.	- Reportes correctos y exportables.- Información coherente con registros de actividades e inventario.

## 6.1.2 Descripción de las Historias de Usuario

Tabla 4. Historio de Usuario 1

<b>NOMBRE DE LA HISTORIA</b>	<b>Registro de Usuario</b>
<b>ID HISTORIA</b>	<b>HU-001</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	<b>Usuario nuevo</b>
<b>Quiero:</b>	<b>Registrarme en la aplicación con mis datos personales</b>
<b>Para:</b>	<b>Poder acceder al sistema y utilizar sus funcionalidades de gestión agrícola.</b>
<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	<b>El usuario ingresa nombre, correo y contraseña válidos</b>
<b>Espero:</b>	<b>Que el sistema valide los datos, cree el usuario y muestre mensaje de confirmación</b>
<b>Cuando:</b>	<b>El usuario omite campos o usa datos inválidos</b>
<b>Espero:</b>	<b>Que el sistema muestre mensajes de error específicos y no permita continuar</b>

Tabla 5. Historia de Usuario 2

<b>NOMBRE DE LA HISTORIA</b>	<b>Iniciar sesión de forma segura</b>
<b>ID HISTORIA</b>	<b>HU-002</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	<b>Usuario registrado</b>
<b>Quiero:</b>	<b>Acceder a la aplicación de forma segura con mis credenciales</b>
<b>Para:</b>	<b>Gestionar mi granja y ver mi información.</b>
<b>CRITERIOS DE ACEPTACIÓN: Autenticación exitosa</b>	

<b>Cuando:</b>	<b>Usuario ingresa usuario y contraseña correctos</b>
<b>Espero:</b>	<b>Validar credenciales y mostrar la pantalla principal</b>
<b>Cuando:</b>	<b>Usuario ingresa credenciales incorrectas,</b>
<b>Espero:</b>	<b>Mostrar mensaje “Correo o contraseña incorrectos” y no permitir acceso</b>

Tabla 6. Historia de Usuario 3

<b>NOMBRE DE LA HISTORIA</b>	<b>Recuperar contraseña</b>
<b>ID HISTORIA</b>	<b>HU-003</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>3</b>
<b>Como:</b>	<b>Usuario registrado que olvidó su contraseña</b>
<b>Quiero:</b>	<b>Recuperar mi contraseña en caso de olvidarla</b>
<b>Para:</b>	<b>Volver a acceder al sistema sin crear una nueva cuenta</b>
<b>CRITERIOS DE ACEPTACIÓN: Solicitud valida</b>	
<b>Cuando:</b>	<b>Solicito recuperación con un correo válido</b>
<b>Espero:</b>	<b>Recibir un enlace para restablecer la cuenta. y poder restablecer</b>
<b>Cuando:</b>	<b>Usuario ingresa un correo no registrado</b>
<b>Espero:</b>	<b>Que el sistema muestra mensaje “Correo no encontrado” y no envíe enlace</b>

Tabla 7. Historia de Usuario 4

<b>NOMBRE DE LA HISTORIA</b>	<b>Cerrar sesión</b>
<b>ID HISTORIA</b>	<b>HU-004</b>

<b>PUNTOS DE LA HISTORIA</b>	<b>3</b>
<b>Como:</b>	<b>Usuario autenticado</b>
<b>Quiero:</b>	<b>Cerrar mi sesión en la aplicación de manera segura</b>
<b>Para:</b>	<b>Proteger mis datos y evitar accesos no autorizados en mi dispositivo</b>
<b>CRITERIOS DE ACEPTACIÓN: Cierre exitoso</b>	
<b>Cuando:</b>	<b>El usuario selecciona la opción “Cerrar sesión”</b>
<b>Espero:</b>	<b>Que el sistema finalice la sesión, borre las credenciales temporales y muestre la pantalla de inicio</b>
<b>Cuando:</b>	<b>El sistema no logra cerrar sesión por error interno</b>
<b>Espero:</b>	<b>Que se muestre un mensaje “No se pudo cerrar sesión, inténtelo nuevamente”</b>

Tabla 8. Historia de Usuario 5

<b>NOMBRE DE LA HISTORIA</b>	<b>Registrar granja</b>
<b>ID HISTORIA</b>	<b>HU-005</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	<b>Usuario registrado</b>
<b>Quiero:</b>	<b>Registrar la información de mi granja</b>
<b>Para:</b>	<b>Gestionar sus datos y actividades</b>
<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	<b>El usuario ingresa todos los datos requeridos</b>
<b>Espero:</b>	<b>Que el sistema cree la granja y la muestre en el panel principal</b>
<b>Cuando:</b>	<b>El usuario omite campos obligatorios o usa datos inválidos</b>

<b>Espero:</b>	<b>Que el sistema muestre mensajes de error y no permita guardar</b>
----------------	--

Tabla 9. Historia de usuario 6

<b>NOMBRE DE LA HISTORIA</b>	<b>Crear lotes dentro de una granja</b>
<b>ID HISTORIA</b>	<b>HU-006</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	Usuario con granja registrada
<b>Quiero:</b>	Crear lotes con nombre, área y tipo de cultivo
<b>Para:</b>	Llevar control detallado de cada espacio productivo
<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	El usuario selecciona una granja y completa los datos del lote
<b>Espero:</b>	Que el sistema relacione el lote con la granja y lo muestre
<b>Cuando:</b>	El usuario no selecciona granja o deja campos vacíos
<b>Espero:</b>	Que el sistema muestre advertencias y no cree el lote

Tabla 10. Historia de Usuario 7

<b>NOMBRE DE LA HISTORIA</b>	<b>Registrar inventario de recursos</b>
<b>ID HISTORIA</b>	<b>HU-007</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	Usuario con lotes activos
<b>Quiero:</b>	Registrar insumos
<b>Para:</b>	Planificar actividades agrícolas con base en recursos reales

<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	El usuario ingresa tipo de recurso, cantidad y unidad
<b>Espero:</b>	Que el sistema almacene el inventario y permita editarlo
<b>Cuando:</b>	El usuario ingresa datos incompletos o inconsistentes
<b>Espero:</b>	Que el sistema muestre errores y no guarde el recurso

Tabla 11. Historia de Usuario 8

<b>NOMBRE DE LA HISTORIA</b>	<b>Registrar actividad agrícola</b>
<b>ID HISTORIA</b>	<b>HU-008</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	Usuario con lotes e inventario registrado
<b>Quiero:</b>	Registrar actividades como siembra, riego o cosecha
<b>Para:</b>	Llevar un historial productivo
<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	El usuario selecciona lote, tipo de actividad
<b>Espero:</b>	Que el sistema guarde la actividad y la relacione con el lote
<b>Cuando:</b>	El usuario no selecciona lote o deja campos vacíos
<b>Espero:</b>	Que el sistema muestre advertencias y no registre la actividad

Tabla 12. Historia de Usuario 9

<b>NOMBRE DE LA HISTORIA</b>	<b>Generar informes descargables</b>
<b>ID HISTORIA</b>	<b>HU-009</b>

<b>PUNTOS DE LA HISTORIA</b>	5
<b>Como:</b>	Usuario con datos registrados
<b>Quiero:</b>	Generar informes en Excel con mis actividades y resultados
<b>Para:</b>	Compartirlos con otros usuarios o instituciones, o tomar decisiones
<b>CRITERIOS DE ACEPTACIÓN: Generación exitoso</b>	
<b>Cuando:</b>	El usuario selecciona el tipo de informe y rango de fechas
<b>Espero:</b>	Que el sistema genere un archivo Excel con los datos filtrado
<b>Cuando:</b>	El usuario no selecciona filtros o hay errores en los datos
<b>Espero:</b>	Que el sistema muestre advertencia y no genere el archivo

Tabla 13. Historia de Usuario 10

<b>NOMBRE DE LA HISTORIA</b>	Módulo institucional
<b>ID HISTORIA</b>	HU-010
<b>PUNTOS DE LA HISTORIA</b>	5
<b>Como:</b>	Usuario Admin
<b>Quiero:</b>	Registrar y visualizar
<b>Para:</b>	Controlar las granjas asociadas y sus datos clave
<b>CRITERIOS DE ACEPTACIÓN: Registro exitoso</b>	
<b>Cuando:</b>	El usuario ingresa nombre, tipo y datos de la entidad
<b>Espero:</b>	Que el sistema cree la secretaría y la relacione con las granjas
<b>Cuando:</b>	El usuario accede al módulo institucional
<b>Espero:</b>	Que se muestren las entidades con sus datos y granjas asociadas

Tabla 14. Historia de Usuario 11

<b>NOMBRE DE LA HISTORIA</b>	<b>Pruebas de rendimiento y carga</b>
<b>ID HISTORIA</b>	<b>HU-011</b>
<b>PUNTOS DE LA HISTORIA</b>	<b>5</b>
<b>Como:</b>	<b>Administrador del sistema</b>
<b>Quiero:</b>	<b>Validar el rendimiento de la aplicación con múltiples usuarios simultáneos</b>
<b>Para:</b>	<b>Asegurar que el sistema funcione correctamente bajo carga real</b>
<b>CRITERIOS DE ACEPTACIÓN: Prueba exitosa</b>	
<b>Cuando:</b>	<b>Se simulan múltiples usuarios accediendo y realizando operaciones</b>
<b>Espero:</b>	<b>Que el sistema mantenga tiempos de respuesta aceptables y sin errores críticos</b>
<b>Cuando:</b>	<b>El sistema se ralentiza o presenta fallos bajo carga</b>
<b>Espero:</b>	<b>Que se registren los errores y se generen reportes para corrección</b>

## 6.2 Arquitectura y Diseño

La arquitectura del sistema se fundamentó en el patrón Model-View-ViewModel (MVVM), con el propósito de garantizar una separación clara entre la interfaz de usuario, la lógica de presentación y los datos. Esta decisión permitió construir una aplicación modular, escalable y mantenible, facilitando la incorporación de nuevas funcionalidades sin comprometer la estabilidad del código existente.

### 6.2.2 Arquitectura MVVM

La implementación de la arquitectura MVVM (Model-View-ViewModel) en el desarrollo de esta aplicación agrícola representa una decisión estratégica orientada a mejorar la organización, escalabilidad y mantenibilidad del sistema. Dado que la plataforma está

diseñada para atender diversos usuarios, MVVM permite establecer una separación clara entre la lógica de negocio, la interfaz de usuario y el modelo de datos.

Gracias a esta arquitectura, es posible definir vistas específicas para cada tipo de usuario, controlando el acceso a la información según sus permisos. Los granjeros pueden gestionar únicamente los datos relacionados con sus propias granjas, mientras que las secretarías de agricultura tienen acceso global a la información, sin capacidad de modificar los registros ingresados por los usuarios. Esta lógica se encapsula en los ViewModels, que actúan como intermediarios entre la interfaz y los modelos, garantizando que cada acción del usuario se traduzca en comportamientos seguros y coherentes.

Además, MVVM facilita la implementación de funcionalidades como la visualización en tiempo real de variables agrícolas (temperatura, humedad, etc.), el control de acceso por roles, y la integración de servicios externos como sensores o APIs meteorológicas. Su estructura modular permite que el sistema evolucione fácilmente, incorporando nuevos módulos o tipos de usuario sin comprometer la estabilidad del código existente.

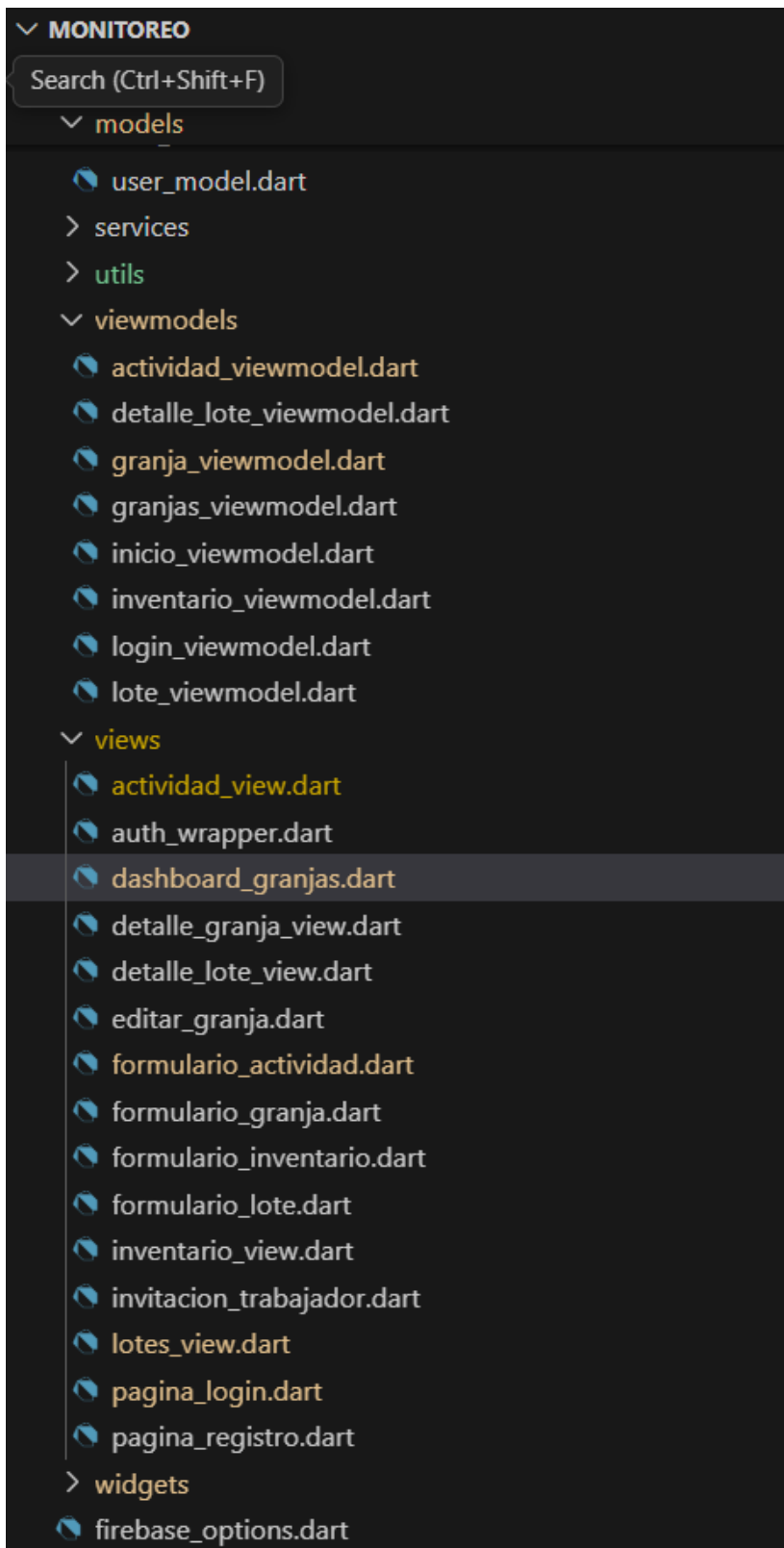


Figura 2. MVVM en flutter

## 6.2.3 Diseño de Base de datos

### Modelo conceptual (entidades principales)

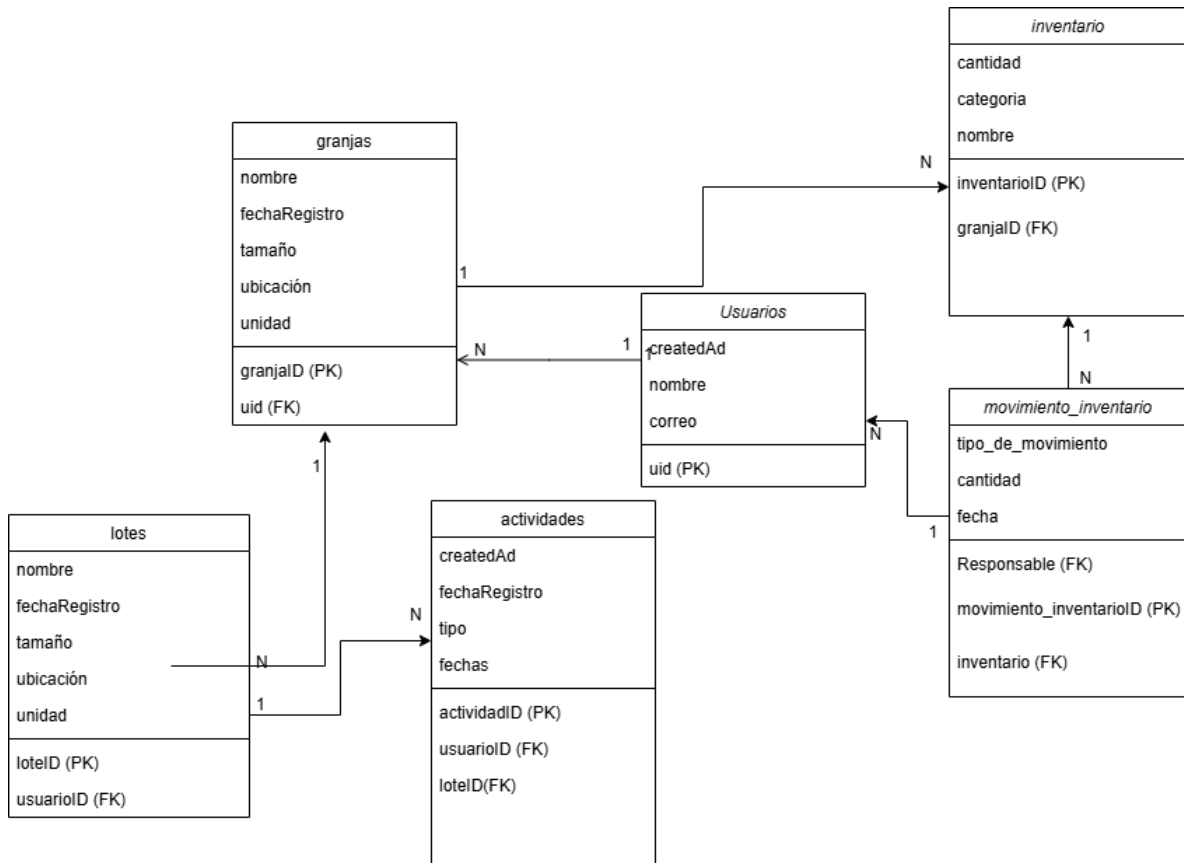


Figura 3. Diagrama entidad relación granjas

## 6.3 Implementación

### Introducción del lenguaje y herramientas utilizadas

El desarrollo de aplicaciones móviles modernas requiere lenguajes y frameworks que permitan construir soluciones rápidas, eficientes y multiplataforma. En este contexto, Dart y Flutter se han consolidado como tecnologías clave para la creación de interfaces dinámicas y de alto rendimiento.

Dart es un lenguaje de programación creado por Google, orientado a objetos y con tipado estático. Flutter, también desarrollado por Google, es un framework que permite construir aplicaciones multiplataforma (Android, iOS, web y escritorio) desde una única base de

código. Su motor gráfico Skia garantiza interfaces fluidas y consistentes, mientras que su enfoque declarativo basado en widgets simplifica la construcción de interfaces de usuario. Una de sus características más destacadas es el Hot Reload, que posibilita visualizar cambios en tiempo real durante el desarrollo, acelerando la iteración y la productividad.

Para la implementación del patrón arquitectónico MVVM (Model–View–ViewModel), Flutter ofrece diversas herramientas de gestión de estado como Provider, Riverpod o Bloc, que permiten separar la lógica de negocio de la presentación y mantener un código más organizado y escalable.

- **Model:** contiene la lógica de negocio y el acceso a datos.
- **View:** representa la interfaz gráfica construida con widgets.
- **ViewModel:** actúa como intermediario, gestionando el estado y exponiendo datos listos para la vista. (Cruz, 2024)

En cuanto a las herramientas de desarrollo, se utiliza Visual Studio Code (VS Code) como editor principal, gracias a su ligereza, extensibilidad y soporte para extensiones específicas de Flutter y Dart. VS Code integra terminales y herramientas de control de versiones como Git, lo que facilita la gestión de ramas, commits y sincronización con repositorios remotos en plataformas como GitHub. Finalmente, la ejecución y pruebas de la aplicación se realizan en un smartphone físico, lo que permite validar el comportamiento real de la aplicación en cuanto a rendimiento, interacción táctil y experiencia de usuario. (TeachMeIDEA, n.d.)

En las figuras x y xx a continuación se puede observar como se ve la app desarrollada:



Figura 4. Programación registro usuario



Figura 5. CRUD granjas

## 6.4 Pruebas

### 6.4.1 Pruebas Unitarias

Casos de prueba a historias de usuario

```
Future<bool> registrarGranjaExtendida(  
    String nombre,  
    double latitud,  
    double longitud,  
    double tamano,  
    String unidad,  
    String tipoGranja,  
    String propietario,  
    String cedula,  
    String telefono,  
    int edad,
```

```

) async {
  _setError(null);
  _setLoading(true);

  try {
    final uid = _auth.currentUser?.uid;
    if (uid == null) throw Exception('Usuario no autenticado');

    await _db.collection('granjas').add({
      'nombre': nombre,
      'latitud': latitud,
      'longitud': longitud,
      'tamano': tamano,
      'unidad': unidad,
      'tipoGranja': tipoGranja,
      'propietario': propietario,
      'cedula': cedula,
      'telefono': telefono,
      'edad': edad,
      'propietarioUid': uid,
      'createdAt': FieldValue.serverTimestamp(),
    });

    _setLoading(false);
    return true;
  } catch (e) {
    _setLoading(false);
    _setError('Error al registrar: $e');
    return false;
  }
}

```

Figura 6. Pruebas unitarias

Esta función ubicada en granja\_viewmodel.dart

- **Crea un nuevo documento** dentro de la colección **granjas**. Ese documento tendrá un **ID único generado automáticamente por Firestore**.
- Los campos (**nombre**, **latitud**, **longitud**, etc.) se guardan como pares clave-valor dentro del documento.

- El campo **createdAt** se llena con la hora del servidor en el momento de la creación.

### 6.3.5 Casos de pruebas

Tabla 15. Caso de prueba\_InicioSesión

<b>Caso de prueba #01</b>	<b>Código del CP</b>	<b>1</b>
	<b>¿Prueba de despliegue?</b>	<b>Si</b>
<b>Descripción</b>	Inicio de sesión.	
<b>Autor de la prueba</b>	David Sierra	
<b>Objetivo de la prueba</b>	Iniciar sesión con credenciales de usuario existentes y válidas.	
<b>Prerrequisitos</b>	<p>* La app debe estar operativo y accesible.</p> <p>* Debe existir al menos un usuario con credenciales válidas en la base de datos (Se cargan por migración a la base de datos).</p>	
<b>Datos de entrada</b>	<p>Email: david.sierra901@pascualbravo.edu.co</p> <p>Contraseña: password</p>	
<b>Pasos</b>	<ol style="list-style-type: none"> <li>1. Navegar a la URL de inicio de sesión (/login), se accede a esta por el botón "Inicio sesión" ubicado en la parte inferior, al centro de la app</li> <li>2. Ingresar el email y la contraseña correctos en los campos correspondientes.</li> <li>3. Hacer clic en el botón de "Iniciar Sesión".</li> <li>4. Dar clic en el botón "Regresar".</li> <li>5. Tratar de acceder con credenciales inválidas.</li> </ol>	
<b>Resultado esperado</b>	<p>La sesión debe iniciarse exitosamente.</p> <p>El usuario debe ser redirigido al dashboard principal de la aplicación.</p> <p>El sistema debe reconocer al usuario como autenticado.</p> <p>Al dar en el botón "Regresar" debe cancelar el inicio de sesión y regresar a la pantalla de Inicio.</p>	

	Al tratar de ingresar con credenciales inválidas tanto el correo como la contraseña, el sistema no debe permitir ingresar y muestra el mensaje “Estas credenciales no coinciden”
Resultado obtenido	El proceso de login funciona según las especificaciones indicadas, hace el proceso y redirige a la vista principal o dashboard. Al dar en el botón "Regresar" cancela la operación de login exitosamente y redirige a la página de inicio. Al tratar de iniciar sesión con las credenciales inválidas el sistema genera la alerta de “credenciales no coinciden” y no permite el acceso.

Tabla 16. Caso prueba\_CRUD Granja

<b>Caso de prueba #02</b>	<b>Código del CP</b>	<b>2</b>
	<b>¿Prueba de despliegue?</b>	<b>Si</b>
<b>Descripción</b>	Creación de una granja con datos válidos.	
<b>Autor de la prueba</b>	David Sierra	
<b>Objetivo de la prueba</b>	Validar que el formulario de granjas permita registrar una granja correctamente y que se muestre en el dashboard.	
<b>Prerrequisitos</b>	*El usuario debe estar autenticado. * La app debe estar operativa y accesible.	
<b>Datos de entrada</b>	Nombre: Granja La Esperanza	

	Ubicación: - Lat: 6.25, Lng: -75.56
	Propietario: David Sierra
	Cédula: 1035876
	Tamaño: 5
	Unidad: Hectáreas
	Tipo de granja: Agrícola
<b>Pasos</b>	<ol style="list-style-type: none"> <li>1. Navegar al dashboard principal.</li> <li>2. Hacer clic en el botón “Agregar granja”.</li> <li>3. Completar todos los campos del formulario con los datos de entrada.</li> <li>4. - Hacer clic en “Guardar”.</li> </ol>
<b>Resultado esperado</b>	<p>La granja se registra exitosamente en la base de datos.</p> <p>El sistema muestra La nueva granja en El listado del dashboard con toda su información.</p>
<b>Resultado obtenido</b>	<p>La nueva granja se guarda con éxito</p> <p>El dashboard refleja la nueva granja con los datos ingresados.</p>

Tabla 17. Caso de prueba\_ Creación lote

<b>Caso de prueba #03</b>	<b>Código del CP</b>	<b>3</b>
	<b>¿Prueba de despliegue?</b>	<b>Si</b>
<b>Descripción</b>	Creación de un lote dentro de una granja existente.	

<b>Autor de la prueba</b>	David Sierra
<b>Objetivo de la prueba</b>	Validar que el sistema permite registrar un lote correctamente dentro de una granja ya creada.
<b>Prerrequisitos</b>	<p>El sistema web debe estar operativo y accesible.</p> <p>El usuario debe estar autenticado.</p> <p>Debe existir al menos una granja registrada en la base de datos.</p>
<b>Datos de entrada</b>	<p>Nombre: “Lote A”</p> <p>Propietario: David Sierra</p> <p>Tamaño:20</p> <p>Unidad: Hectáreas</p>
<b>Pasos</b>	<ol style="list-style-type: none"><li>1. Navegar al dashboard principal.</li><li>2. Seleccionar una granja existente tocando sobre su card.</li><li>3. Hacer clic en el botón “Lotes”.</li><li>4. Presionar el botón flotante inferior con el ícono “+” para agregar un lote.</li><li>5. Completar los campos requeridos: nombre, tamaño y unidad de medida.</li><li>6. Click en “Guardar lote”.</li></ol>

---

**Resultado  
esperado**

El lote se registra exitosamente en la base de datos asociado a la granja seleccionada.

---

El sistema muestra el nuevo lote en la vista de detalle de la granja.

---

El usuario permanece autenticado y puede visualizar la información del lote recién creado.

---

**Resultado  
obtenido**

El lote se crea con éxito.

---

La información se almacena correctamente en la base de datos vinculada a la granja.

---

El dashboard de la granja refleja el nuevo lote con los datos ingresados.

---

## **6. Conclusiones**

El desarrollo de la aplicación multiplataforma para la gestión agrícola en granjas antioqueñas permitió demostrar que la digitalización del sector agropecuario es una estrategia viable y necesaria para mejorar la eficiencia, la productividad y la trazabilidad de los procesos productivos. La implementación de la arquitectura MVVM, junto con el uso de Flutter y Firebase Firestore, da una solución moderna, escalable y accesible, capaz de adaptarse a diferentes perfiles de usuario y contextos de producción.

La metodología ágil SCRUM fue fundamental para organizar el trabajo en ciclos iterativos, validar constantemente las funcionalidades con los usuarios y ajustar el sistema a las necesidades reales del campo. Gracias a este enfoque, se logró entregar un producto funcional en cada sprint, fortaleciendo la comunicación con los agricultores y asegurando la pertinencia de la herramienta.

En conclusión, este trabajo constituye un aporte significativo a la transformación digital del agro antioqueño, ofreciendo una herramienta práctica y adaptable que mejora la calidad de vida de los agricultores y sienta las bases para futuras ampliaciones orientadas a la agricultura de precisión y la integración de tecnologías emergentes.

## 7. Recomendaciones

El proyecto desarrollado constituye una base sólida para la digitalización de las granjas antioqueñas; sin embargo, se identifican oportunidades de mejora que pueden potenciar aún más su impacto en el sector agropecuario:

- Automatización de procesos agrícolas: añadir módulos que sugieran automáticamente actividades como riego, fertilización o rotación de cultivos, basados en datos históricos y condiciones ambientales.
- Conexión con sensores IoT: integrar dispositivos de monitoreo en campo (temperatura, humedad, calidad del suelo) para obtener información en tiempo real y mejorar la precisión de los registros.
- Soporte offline: desarrollar mecanismos que permitan registrar información sin conexión a internet y sincronizar los datos automáticamente cuando se restablezca la conectividad.
- Reportes avanzados y visualización de datos: implementar dashboards interactivos que permitan analizar tendencias, comparar lotes y evaluar indicadores de productividad de manera gráfica y dinámica.

## 8. Referencias Bibliográficas

Colombia Aprende. (2021). *Cartilla del sector agropecuario*. Ministerio de Educación Nacional.

[https://www.colombiaaprende.edu.co/sites/default/files/files\\_public/2021-08/cartilla-sector-agropecuario.pdf](https://www.colombiaaprende.edu.co/sites/default/files/files_public/2021-08/cartilla-sector-agropecuario.pdf)

DesarrolloLibre. (2023). *Arquitectura MVVM en Flutter: mejores prácticas con Provider y HTTP*.

Recuperado de <https://www.desarrollolibre.net/blog/flutter/arquitectura-mvvm-en-flutter-mejores-practic-as-con-provider-y-http>

TeachMeIDEA. (2024). *How to implement MVVM architecture in Flutter*. Recuperado de

<https://teachmeidea.com/how-to-implement-mvvm-architecture-in-flutter/>

(n.d.). Agroptima: Software y Aplicación de gestión agrícola fácil de usar. Retrieved April

23, 2025, from <https://www.agroptima.com/>

*Isagri*. (n.d.). Isagri - Digitalización Agricultura, Ganadería y Bodegas. Retrieved April 23,

2025, from <https://www.isagri.es/>

Ovalle Másmela, J., Romero-Perdomo, F., & Uribe Galvis, C. (2023). *Tecnologías emergentes para el agro y su aplicación en Colombia*. Corporación Colombiana de Investigación

Agropecuaria (AGROSAVIA). <https://doi.org/10.21930/agrosavia.estudiodevigilancia.2023.2>