

**DOCUMENTO DE ARQUITECTURA DE SOFTWARE:**  
**Agriculture Company**

**Universidad Tecnológica de Bolívar**

**DOCUMENTO DE**  
**ARQUITECTURA DE SOFTWARE:**  
*Agriculture Company*

***Presentado por:***

Nicolas Molina Díaz

Diego Martínez Lora

Luis Carlos Pacheco

**6/11/2024**

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

# TABLA DE CONTENIDO

1. Introducción.....	3
1.1. Propósito.....	3
1.2. Ámbitos del sistema.....	3
1.3. Definiciones, siglas y abreviaturas.....	3
2. Macro Arquitectura.....	3
2.1. Descripción del Sistema .....	3
- Módulo de Cultivo.....	4
- Módulo de Inventario .....	4
2.2. Vista de Arquitectura.....	4
3. Arquitectura de Componentes .....	5
4. Diseño de Base de Datos .....	6
4.1. Modelado de datos.....	6
4.2. Integridad y escalabilidad .....	8
5. Vista de Interfaces .....	9
5.1. Interfaces de Usuario .....	9
5.2. Interfaces de Hardware y Software.....	9
5.3. Interfaces de Comunicación.....	9
6. Requisitos No Funcionales .....	10
6.1. Requerimientos de Desempeño .....	10
6.2. Requerimientos de Seguridad .....	10
6.3. Atributos de Calidad del Software.....	10
7. Diagrama de Arquitectura.....	11
7.1. Diagrama de clases o componentes .....	11
7.2. Diagrama de procesos.....	12
7.3. Diagrama de casos de uso.....	13
8. Conclusión.....	14

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

### 1. Introducción

#### 1.1. Propósito

El propósito de este documento es definir la arquitectura de software para el sistema Agriculture Company, una solución avanzada que optimiza procesos clave en la agricultura, tales como la gestión de cultivos, inventario, distribución y ventas. Esta arquitectura está diseñada para soportar las herramientas y funcionalidades necesarias para mejorar la productividad, eficiencia y rentabilidad en la cadena de valor agrícola.

#### 1.2. Ámbitos del sistema

El sistema Agriculture Company está orientado a transformar la gestión agrícola mediante tecnologías de vanguardia que faciliten la toma de decisiones en tiempo real. A través de módulos especializados, este sistema busca optimizar todas las fases de la producción agrícola, desde el monitoreo de cultivos hasta la gestión de inventarios y la coordinación en la distribución de productos, beneficiando explotaciones agrícolas de todos los tamaños.

#### 1.3. Definiciones, siglas y abreviaturas

- **HTTP/HTTPS (Protocolo de Transferencia de Hipertexto/Protocolo de Transferencia de Hipertexto Seguro):** HTTP es el protocolo que permite la transferencia de datos a través de la web, facilitando la comunicación entre un navegador y un servidor. HTTPS es la versión segura de HTTP; encripta la información mediante SSL/TLS para asegurar la confidencialidad e integridad de los datos transmitidos.
- **PostgreSQL:** Sistema de gestión de bases de datos relacional de código abierto que utiliza y amplía el lenguaje SQL, permitiendo un manejo avanzado de datos y la posibilidad de personalización mediante extensiones.
- **Docker:** Plataforma de contenedorización de software de código abierto que permite a los desarrolladores empaquetar aplicaciones y todas sus dependencias en contenedores ligeros y portátiles. Los contenedores Docker aseguran que las aplicaciones se ejecuten de manera consistente en cualquier entorno, desde el desarrollo hasta la producción.
- **Docker Compose:** Herramienta de Docker que permite definir y ejecutar aplicaciones de múltiples contenedores a través de un solo archivo de configuración, generalmente llamado docker-compose.yml. Docker Compose facilita el despliegue y la administración de aplicaciones complejas compuestas por varios servicios, como bases de datos, aplicaciones backend y servidores web, permitiendo su inicio simultáneo y orquestado con un solo comando.
- **Alembic:** Herramienta de migración de bases de datos para Python, diseñada para trabajar con SQLAlchemy, un ORM (Object-Relational Mapping) popular. Alembic permite gestionar los cambios en el esquema de una base de datos de manera controlada y versionada, facilitando la creación, aplicación y reversión de migraciones en entornos de desarrollo y producción.

### 2. Macro Arquitectura

#### 2.1. Descripción del Sistema

En el sistema Agriculture Company, los módulos de cultivo, inventario, distribución y ventas están integrados para facilitar un flujo continuo de datos a lo largo de la cadena de valor agrícola. A continuación, se describe cómo interactúan estos módulos:

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

### - Módulo de Cultivo

Este módulo permite a los usuarios monitorear el estado de los cultivos y gestionar recursos necesarios para su crecimiento, como agua, fertilizantes y pesticidas.

Los datos generados sobre el rendimiento y la calidad de los cultivos se envían al módulo de inventario para actualizar la disponibilidad de productos futuros.

También se conectan con el módulo de distribución para planificar las necesidades logísticas en función de los tiempos de cosecha, maximizando la frescura y calidad de los productos.

### - Módulo de Inventario

El módulo de inventario recibe información del módulo de cultivo para realizar estimaciones de insumos y productos que estarán disponibles tras la cosecha.

Actualiza los niveles de stock en tiempo real y permite una gestión de insumos eficiente, asegurando que los recursos estén disponibles para futuras actividades de cultivo.

Proporciona datos al módulo de ventas para garantizar que la oferta en el mercado sea precisa y oportuna, ajustando la disponibilidad de productos según los resultados de la producción agrícola.

## 2.2. Vista de Arquitectura

Figura 1. Diagrama de actividades para caso de uso administrador

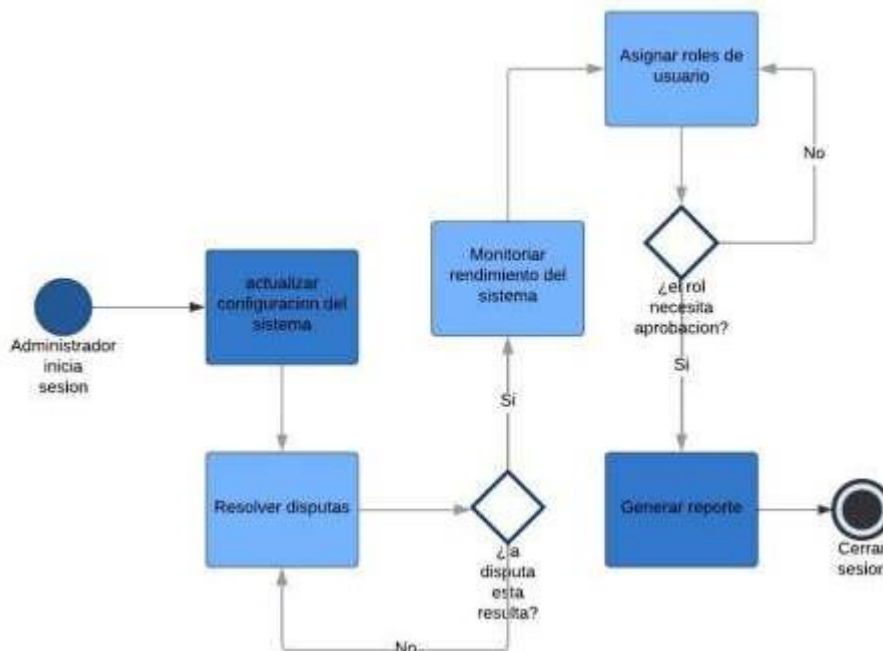
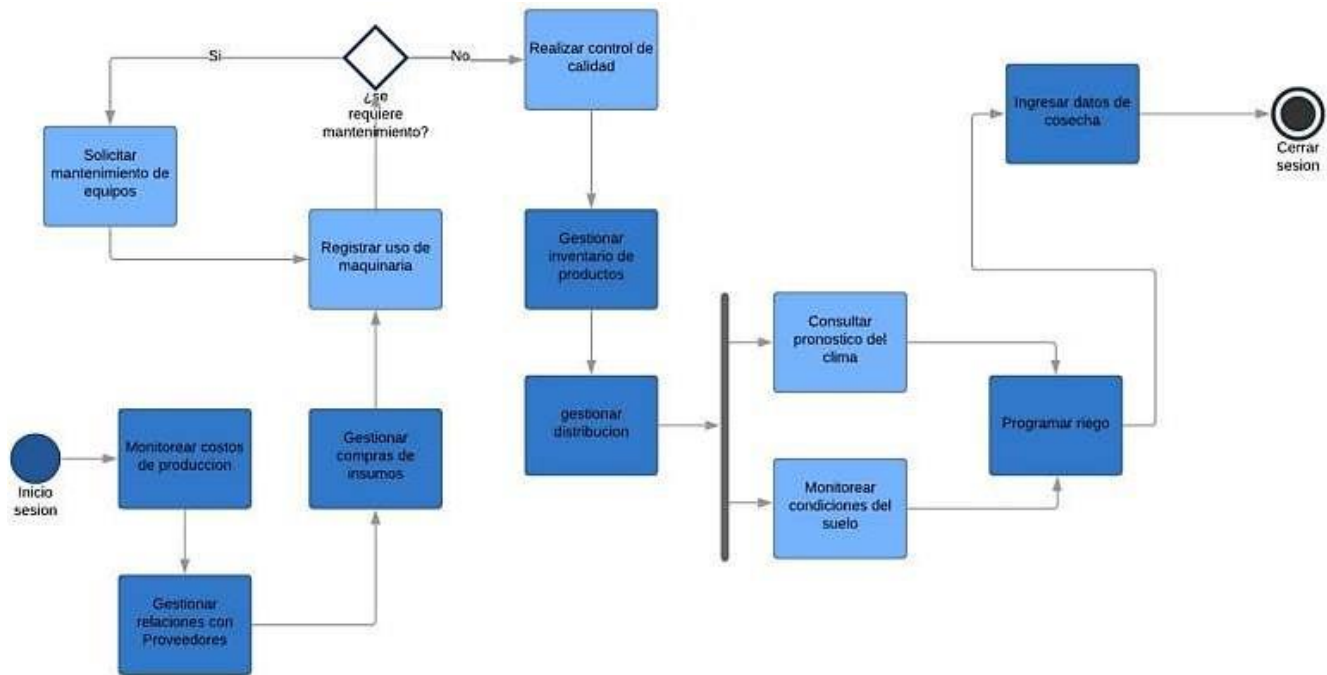


Figura 2. Diagrama de actividades para caso de uso granjero.

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company



### 3. Arquitectura de Componentes

- **Gestión de Cultivos**

- **Objetivo:**

Dar soporte a la planificación de silos y cultivos disponibles, asegurando que el usuario pueda realizar una gestión segura y efectiva de las áreas de cultivo y optimizar su uso.

- **Funcionalidades:**

**Planificación y Seguridad de Datos:**

Asegurar que el usuario pueda verificar fácilmente qué cultivos y silos están disponibles, evitando superposiciones o mal uso del terreno.

**Implementación de IA para Recomendaciones:**

Aunque está prevista para fases futuras, se deberá estructurar el sistema para que pueda integrar un modelo de IA basado en patrones climáticos, suelo y datos de enfermedades para recomendar las mejores ubicaciones de cultivos y ciclos de producción.

- **Interacciones con Otros Módulos:**

Con el Inventario: Integrar la información de cosechas planificadas y en crecimiento, lo que permite un control completo de recursos.

Con el Sistema de Clima y Suelo: Para mejorar la calidad de las recomendaciones de IA, este módulo podrá conectarse en fases futuras a datos meteorológicos y análisis de suelo.

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

- **Gestión de Inventario**

- **Objetivo:**

Proporcionar un registro detallado de las cosechas, incluyendo ubicación, fecha de cultivo, estado de crecimiento y cantidad de recursos disponibles en cada silo y almacén.

- **Funcionalidades:**

- Registro Detallado de Cosechas:**

Mantener un registro actualizado de la cantidad de cada cosecha y su ubicación específica, apoyando la planificación de distribución.

- Estado de Crecimiento:**

Permitir a los usuarios rastrear el progreso de cada cultivo, desde su plantación hasta la cosecha, para tener información precisa sobre disponibilidad futura.

- **Interacciones con Otros Módulos:**

Con Gestión de Cultivos: Sincronizar los datos para asegurar que la planificación de nuevos cultivos tome en cuenta la disponibilidad de recursos y el estado de crecimiento actual de cada cultivo.

Con Distribución y Ventas: Facilitar un acceso rápido a los niveles de inventario, permitiendo ver la disponibilidad inmediata para ventas y planificación de distribución.

- **Gestión de Distribución y Ventas**

- **Objetivo:**

Coordinar la distribución y ventas con la disponibilidad de inventario y cultivos, minimizando tiempos de espera y evitando cuellos de botella en la entrega.

- **Funcionalidades:**

- Optimización de Distribución:** Organizar rutas y fechas de entrega según el inventario disponible y los pedidos.

- **Interacciones con Otros Módulos:**

- Con Inventario:** Permitir reservas automáticas de inventario una vez que se confirma una venta, evitando sobreventas y coordinando la disponibilidad.

- Con Gestión de Cultivos:** Sincronizar el calendario de cosechas con el de distribución para garantizar una oferta continua de productos frescos.

## 4. Diseño de Base de Datos

### 4.1. Modelado de datos

Principales Entidades y Atributos:

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

- **Cultivo**

- ID\_Cultivo (PK): Identificador único del cultivo.
- Tipo (String): Tipo de cultivo (por ejemplo, trigo, maíz).
- Área\_cultivada (Float): Superficie de cultivo.
- Fecha\_siembra (Date): Fecha en la que fue sembrado.
- Fecha\_cosecha (Date): Fecha estimada o real de cosecha.
- Estado\_crecimiento (String): Estado actual del cultivo.
- Necesidades\_tratamiento (Text): Descripción de tratamientos específicos necesarios para el cultivo.

- **Cosecha**

- ID\_Cosecha (PK): Identificador único de la cosecha.
- Fecha\_cosecha (Date): Fecha en la que se realizó la cosecha.
- Cantidad\_cosecha (Float): Cantidad cosechada en unidades específicas.
- Área (Float): Área de cultivo de la cosecha.
- ID\_Cultivo (FK): Relación con el cultivo del cual proviene la cosecha.

- **Silo**

- ID\_Silo (PK): Identificador único del silo.
- Nombre (String): Nombre o identificación del silo.
- Capacidad (Float): Capacidad total del silo.
- Contenido (Float): Cantidad actual de producto almacenado en el silo.
- ID\_Cosecha (FK): Relación con la cosecha almacenada en el silo.

- **Punto de Venta**

- ID\_Punto\_Venta (PK): Identificador único del punto de venta.
- Nombre (String): Nombre del punto de venta.
- Dirección (String): Dirección física del punto de venta.

- **Ventas**

- ID\_Venta (PK): Identificador único de la venta.
- Fecha (Date): Fecha en la que se realizó la venta.
- Cantidad\_vendida (Float): Cantidad de producto vendido.
- Precio (Decimal): Precio total de la venta.
- ID\_Punto\_Venta (FK): Relación con el punto de venta donde se realizó la venta.

- **Encargo (Entidad débil)**

- ID\_Encargo (PK): Identificador único del encargo.
- Fecha (Date): Fecha del encargo.
- Cantidad\_producto (Float): Cantidad de producto solicitada en el encargo.
- ID\_Vehículo (FK): Relación con el vehículo asignado para el encargo.
- ID\_Punto\_Venta (FK): Relación con el punto de venta correspondiente.

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

- **Vehículo**

- ID\_Vehículo (PK): Identificador único del vehículo.
- Matrícula (String): Matrícula del vehículo.
- Capacidad\_Carga (Float): Capacidad de carga del vehículo.
- ID\_Cosecha (FK): Relación con la cosecha asignada al vehículo.

Relaciones Entre Entidades:

- **Cultivo - Cosecha:** Relación de uno a muchos (un cultivo puede tener múltiples cosechas a lo largo de las temporadas).
- **Cosecha - Silo:** Relación de uno a muchos (una cosecha puede almacenarse en varios silos).
- **Punto de Venta - Ventas:** Relación de uno a muchos (un punto de venta puede tener varias ventas).
- **Ventas - Encargo:** Relación de uno a muchos (una venta puede estar asociada con múltiples encargos).
- **Encargo - Vehículo:** Relación de muchos a uno (un encargo está asignado a un único vehículo).
- **Vehículo - Cosecha:** Relación de uno a muchos (un vehículo puede estar asignado a una sola cosecha en un momento, pero una cosecha puede tener varios vehículos asignados en distintas entregas).

#### 4.2. Integridad y escalabilidad

- **Escalabilidad:**

- **Particionado Horizontal:** Para mejorar la escalabilidad y el rendimiento, especialmente en tablas grandes como **Cosecha** y **Ventas**, se puede aplicar particionado horizontal por temporadas o ubicaciones geográficas.
- **Optimización de Índices:** Crear índices en las columnas más consultadas (ID\_Cultivo, ID\_Cosecha, Fecha\_cosecha, ID\_Punto\_Venta) para agilizar las consultas y mejorar el rendimiento del sistema al manejar grandes volúmenes de datos.
- **Replicación de Base de Datos:** Configurar replicación en el sistema para balancear la carga y mejorar la disponibilidad de datos, especialmente en entornos de alta demanda.

- **Integridad de Datos:**

- **Integridad Referencial:** Mantener relaciones sólidas mediante claves foráneas, asegurando que las entidades relacionadas tengan consistencia (por ejemplo, cada **Encargo** debe tener un **Punto de Venta** y un **Vehículo** válidos).
- **Restricciones y Validaciones:**
  - Definir reglas para asegurar que cantidades en campos como Cantidad\_cosecha, Cantidad\_vendida y Capacidad\_Carga sean valores positivos.



# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

- Validar que la capacidad en **Silo** nunca exceda su Capacidad.
- Mantener coherencia entre fechas, por ejemplo, Fecha\_siembra de un **Cultivo** debe ser anterior a Fecha\_cosecha.
- **Seguridad:**
  - **Control de Acceso:** Implementar roles y permisos de acceso para proteger los datos confidenciales y limitar las operaciones de lectura, escritura y eliminación a los usuarios autorizados.
  - **Cifrado de Datos Sensibles:** Aplicar cifrado para los datos financieros (ej., Precio en **Ventas**) y cualquier otra información que sea sensible en el sistema.

## 5. Vista de Interfaces

### 5.1. Interfaces de Usuario

- **Pantalla de inicio de sesión:** Una interfaz simple y segura que permite a los usuarios ingresar al sistema utilizando sus credenciales. La autenticación asegura el acceso controlado y la protección de la información.
- **Pantalla de notificaciones:** Diseñada para mostrar notificaciones y recordatorios generados por el sistema. Aunque actualmente esta funcionalidad aún no está implementada, se espera que permita a los usuarios mantenerse informados sobre eventos importantes y tareas pendientes.
- **Pantalla de gestión de cultivos:** Permite a los usuarios visualizar, planificar y actualizar detalles de los cultivos, como la fecha de siembra, el estado de crecimiento, el área cultivada, y cualquier tratamiento necesario. Esta pantalla centraliza la información para facilitar la toma de decisiones.

### 5.2. Interfaces de Hardware y Software

- **Dispositivos soportados:**
  - **PCs y laptops:** El sistema es compatible con computadoras de escritorio y portátiles.
  - **Dispositivos móviles:** El sistema funciona en dispositivos móviles como celulares y tabletas.
- **Sistemas operativos compatibles:**
  - **PCs y laptops:** Compatible con Windows, macOS y Linux.
  - **Dispositivos móviles:** Compatible con las versiones recientes de Android e iOS.
- **Compatibilidad de navegadores:** El sistema es accesible desde los navegadores principales (Google Chrome, Mozilla Firefox, Microsoft Edge y Safari), asegurando que mantenga su funcionalidad completa y un diseño responsivo en todos ellos.

### 5.3. Interfaces de Comunicación

- **Comunicación web:** El sistema utiliza protocolos HTTP/HTTPS para garantizar la seguridad y privacidad en la comunicación de datos. Esto permite a los usuarios interactuar con la plataforma a través de una interfaz web que soporta la programación y gestión de citas y otras funciones administrativas.

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

- **Interacción con otros sistemas:** Además de la comunicación con los usuarios, el sistema está diseñado para integrarse con otras plataformas y sistemas agrícolas en la nube, lo cual facilita la interoperabilidad y la gestión eficiente de datos relacionados con inventarios, cosechas y distribución.

## 6. Requisitos No Funcionales

### 6.1. Requerimientos de Desempeño

- Procesamiento en tiempo real para permitir decisiones rápidas y evitar problemas de stock.
- Tiempos de respuesta rápidos en consultas y actualizaciones de inventario.
- Capacidad para manejar picos de carga durante las temporadas de cosecha.

### 6.2. Requerimientos de Seguridad

- **Autenticación de usuarios** con control de acceso basado en roles (por ejemplo, administradores y clientes).
- **Cumplimiento de normativas de seguridad** aplicables (como ISO).
- **Cifrado de datos** en las comunicaciones entre microservicios y APIs.
- **Registro de auditoría** para todas las transacciones críticas.

### 6.3. Atributos de Calidad del Software

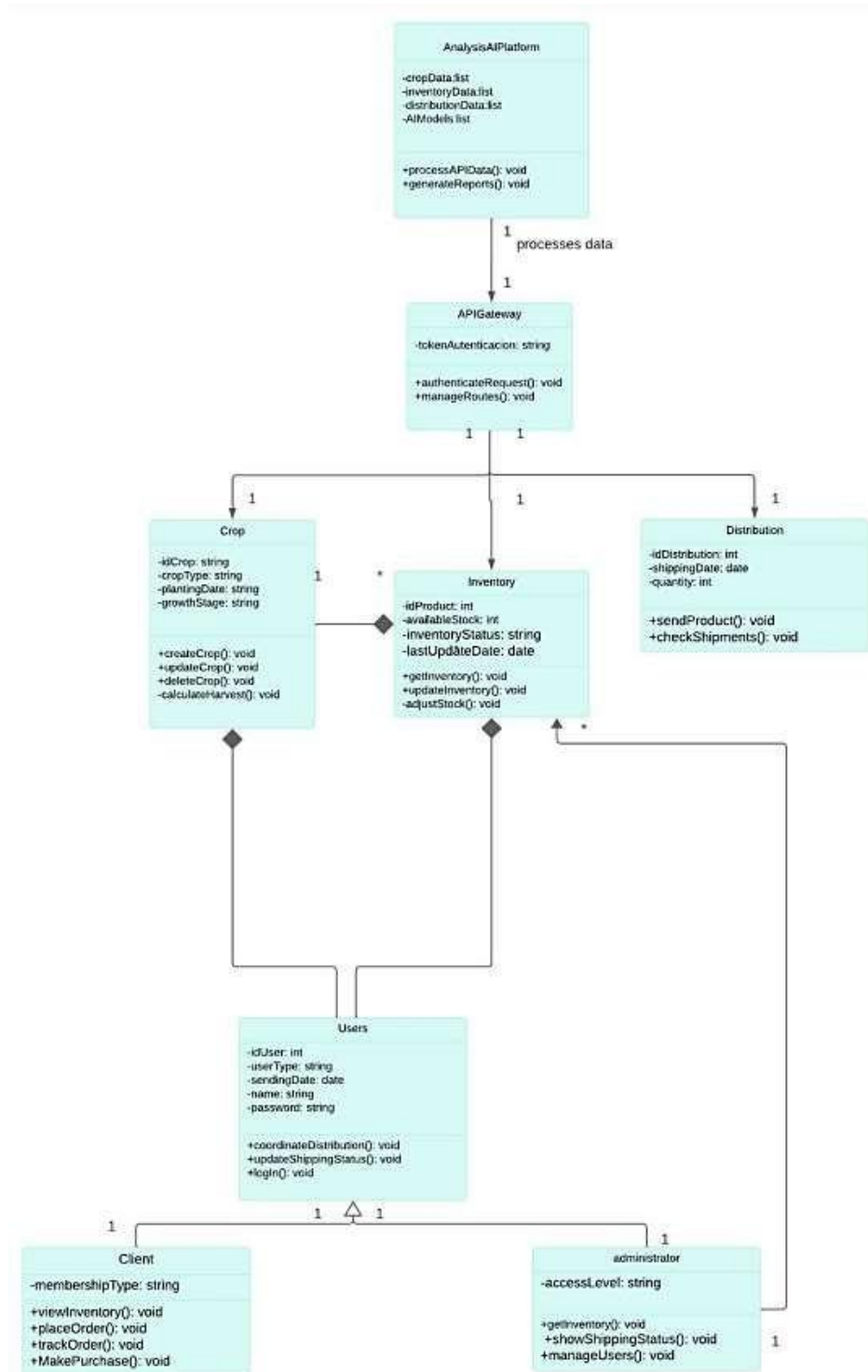
- **Fiabilidad:** El sistema debe garantizar una disponibilidad del 99.9%.
- **Escalabilidad:** Capacidad para manejar un mayor volumen de datos sin pérdida de rendimiento, especialmente en temporadas de cosecha con mayor actividad de usuarios.
- **Mantenibilidad:** Facilidad para actualizar los microservicios sin interrumpir el funcionamiento del sistema.
- **Interoperabilidad:** Integración fluida con otros sistemas agrícolas y plataformas en la nube.
- **Usabilidad:** Interfaces intuitivas para usuarios de diferentes niveles de conocimientos técnicos.

# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

## Agriculture Company

### 7. Diagrama de Arquitectura

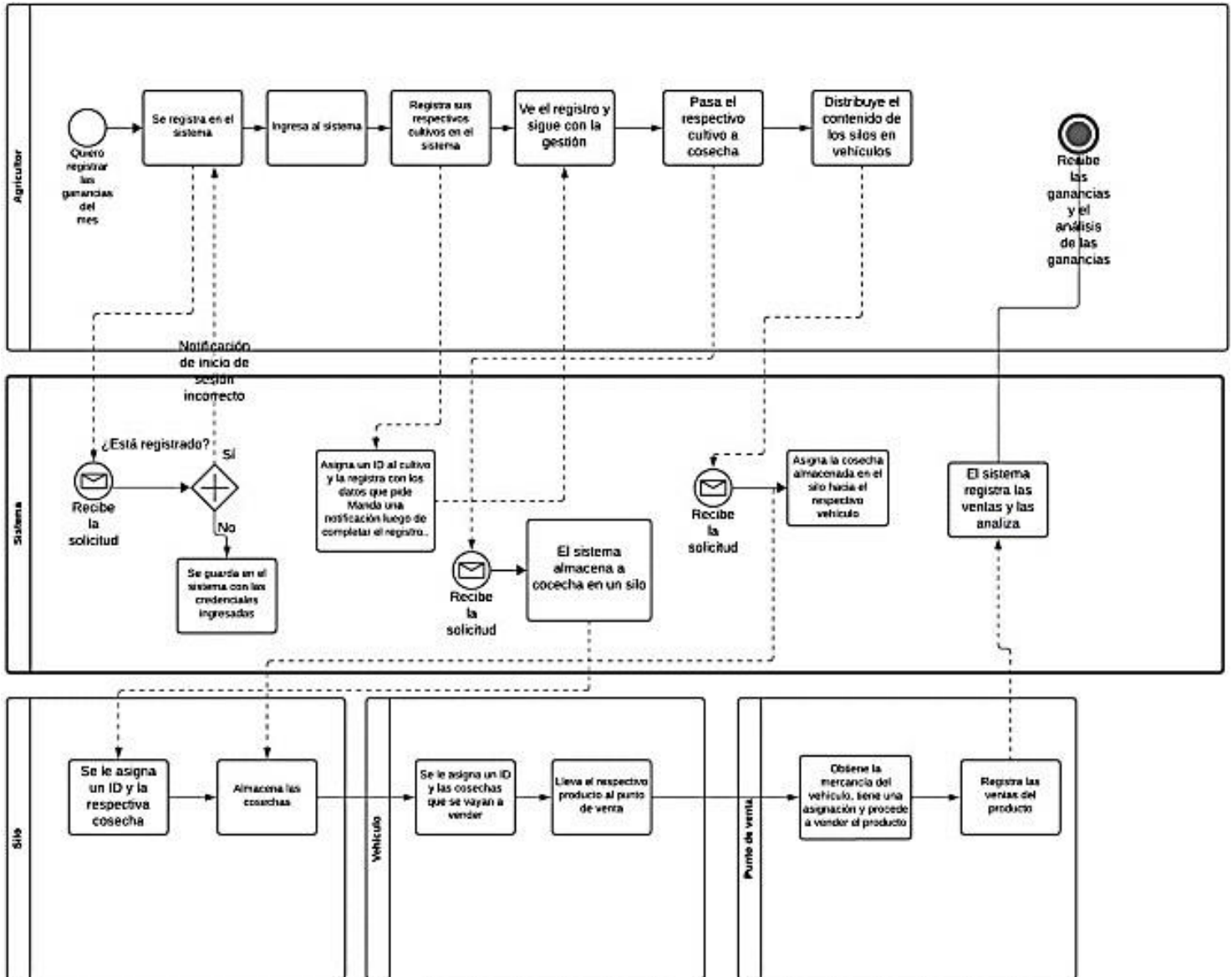
#### 7.1. Diagrama de clases o componentes.



# DOCUMENTO DE ARQUITECTURA DE SOFTWARE:

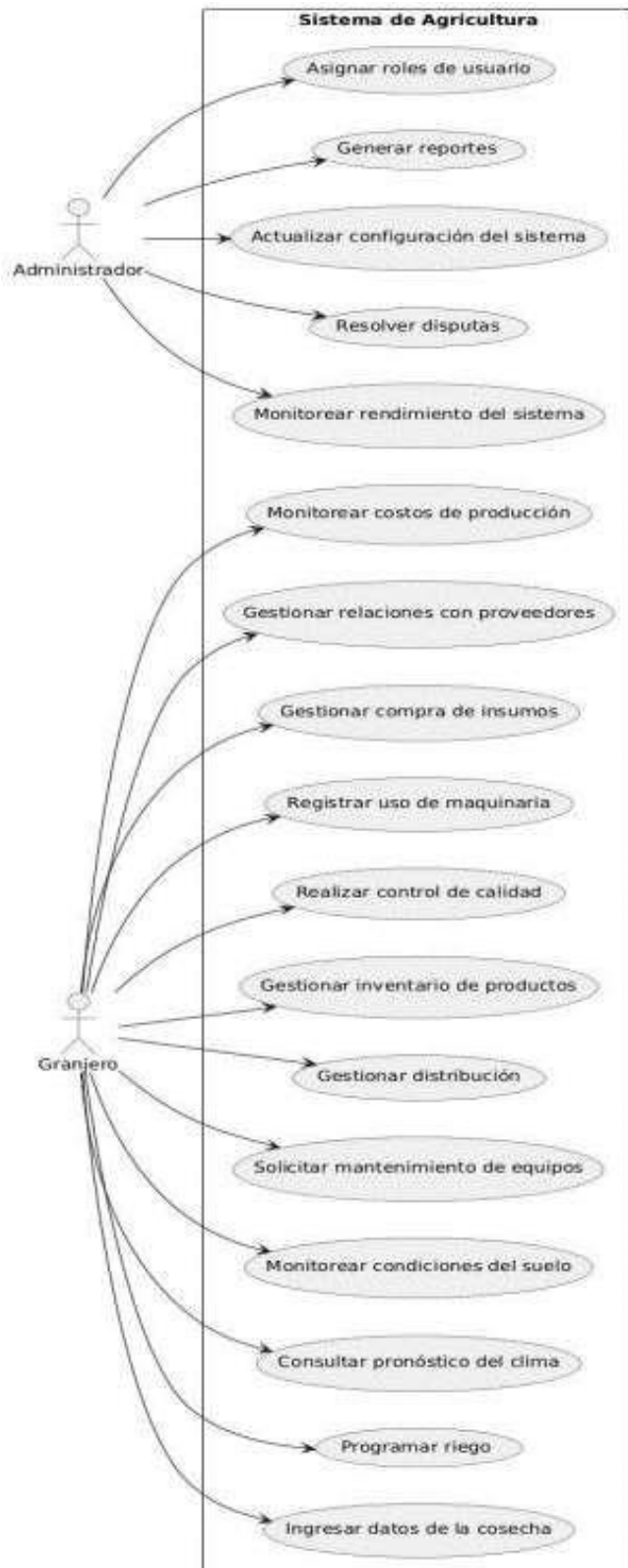
## Agriculture Company

### 7.2. Diagrama de procesos.



# DOCUMENTO DE ARQUITECTURA DE SOFTWARE: Agriculture Company

## 7.3. Diagrama de casos de uso.



# **DOCUMENTO DE ARQUITECTURA DE SOFTWARE:**

## **Agriculture Company**

### **8. Conclusión**

La arquitectura del sistema es adaptable, escalable y basada en datos, diseñada para optimizar la gestión agrícola en múltiples entornos y dispositivos. Su estructura modular permite actualizaciones sin interrupciones, soporta grandes volúmenes de datos y facilita la integración con sistemas externos, asegurando interoperabilidad y seguridad en las operaciones. Las interfaces intuitivas y la capacidad de procesamiento en tiempo real permiten una gestión eficiente de cultivos, inventarios y distribución, proporcionando a los usuarios información precisa y actualizada para una toma de decisiones rápida y efectiva.