**PRESENTACION**

Hola, mi nombre es Marcos Schmid y estoy desarrollando una aplicacion para moviles Android usando lenguaje Kotlin. Tengo conocimientos muy basicos de programacion por lo que he desarrollado hasta el momento lo he hecho con la ayuda de varias inteligencias artificiales distintas. Esta aplicacion la empecé a desarrollar para que me sirva como ayuda en el trabajo que realizo, pero luego de un tiempo pensé en la posibilidad de desarrollar la aplicacion con la posibilidad de monetizarla de alguna manera en algún momento.

**MI TRABAJO**

Tengo una empresa (T&S Agroaplicaciones) dedicada a realizar aplicaciones agricolas con drones DJI de la linea Agras. En este momento la empresa cuenta con dos de estas unidades los cuales pueden se utilizados para realizar tanto pulverizaciones liquidas (herbicidas, insecticidas, fungicidas, etc) como del tipo solidas (siembra, fetilizacion, etc).

La empresa está radicada en la localidad de Chilibroste, Cordoba, Argentina. Trabajamos en toda la zona de influencia.

Somos una empresa del tipo contratista por lo que generalmente realizamos trabajos para terceros. Son estos productores agricolas los que se ponen en contacto con la empresa para realizar un trabajo especifico. Para ello, el productor nos informa principalmente el tipo de aplicacion que desea realizar, la cantidad de hectareas del campo, la ubicacion y los productos que desea utilizar. Una vez confirmado el trabajo se debe buscar un dia con las condiciones climaticas adecuadas para realizar la aplicacion ya que un viento demasiado fuerte puede hacer derivar los productos hacia el campo de un productor vecino y arruinar todo un cultivo que no sea resistente a los productos que se estan aplicando. Ademas de la velocidad del viento se debe considerar la direccion del mismo, la humedad ambiente, la temperatura, el indice KP y demás dificultades que puedan presentarse en la ubicacion del campo a trabajar.

Cuando las condiciones lo permiten nos dirijimos al lugar del trabajo y empezamos con los preparativos previos. Estos incluyen la verificacion tecnica de los drones, carga de baterias, combustible disponible para cargar los generadores electricos encargados de la carga de las baterias de los drones y otras cuestiones tecnicas. Lo mas importante es el mapeado de los lotes que se realiza con uno de los controles del dron y luego se traslada al otro control para que las trayectorias y marcado de obstaculos sean identicos en cada control. Luego se establecen los parametros de aplicacion (caudal en litros o Kg por hectarea, tamaño de gota en micrones, altura, velocidad e interlineado en metros). Estos parametros son iguales para cada dron. De todos modos pueden ser modificados durante el trabajo.

Cuando los drones están listos para la aplicacion se procede a la preparacion de los productos a aplicar. El productor es el que indica los productos que quiere utilizar y la dosis por hectarea que desea aplicar de cada uno de ellos. La mezcla de los productos se realiza en un tanque mezclador (mixer) desde donde luego se abastece a los drones.

El calculo de la receta es un factor muy importante para la correcta realizacion del trabajo. Para ello se debe tener en cuenta la cantidad de hectareas que se podrán trabajar dependiendo de la cantidad de liquidos mezclados (caldo) que se pueden agregar en el mixer y la dosis que el productor desea aplicar por hectarea. Estos productos se van agregando al mixer de forma ordenada segun el tipo de formulacion del producto. Generalmente se completa la cantidad total con agua, se espera un momento a que los liquidos estén bien mezclados y luego se abastece a los drones para comenzar el trabajo.

Una vez terminado el trabajo se limpian todos los equipos y se deja todo preparado para la proxima tarea.

Cada tipo de aplicacion tiene un costo especifico. Por ejemplo, actualmente una aplicacion liquida tiene un costo de 12 dolares por hectarea. Las aplicaciones solidas tienen otro precio y así sucesivamente. Estos valores los establece cada empresa que se dedica al mismo trabajo. El costo de los productos utilizados corre por cuenta del productor.

**OBJETIVOS DE LA APP**

La app está pensada para facilitar todas las tareas realizadas por la empresa de aplicaciones. Por lo tanto es muy importante llevar un registro de clientes, trabajos asociados a cada cliente con todos sus detalles, administracion, calculo de recetas y visualizacion del clima entre otros.

La app cuenta con una barra de navegacion inferior con un boton “+” integrado que responde según la pantalla en la que nos encontremos.

**POSIBLES IMPLEMENTACIONES A DESARROLLAR**

**Plan de trabajo**

El plan de trabajo es un factor muy importante antes de realizar un trabajo ya que esto permite tener todos los aspectos del trabajo a realizar antes de dirigirse al lugar de trabajo. Esto permite reducir de manera considerable el tiempo que se necesita para finalizar el trabajo ya que de esta manera no se pierde tiempo innecesario al momento de la realizar la aplicacion en el campo.

Este plan se realiza de la siguiente manera: en primer lugar hay que saber la ubicacion correcta del lote o lotes donde se hará la aplicacion. Lo ideal es que el perimetro de los lotes sean marcados en la ubicacion fisica de los mismos con el control remoto de la aeronave o incluso con el dron. Una vez que el perimetro del lote o lotes han sido marcados en el control remoto se procede a dividir el lote en partes mas pequeñas para luego segmentar las trayectorias para que cada dron realice su propio recorrido. Generalmente cada aeronave trabaja el 50% de cada lote. Ademas al dividir los lotes en partes mas pequeñas se logra que el equipo reabastecedor pueda ubicarse siempre cerca del sector donde estan trabajando las aeronaves.

Muchas veces no se dispone de muchos luegares disponibles para ubicar el equipo reabastecedor por lo que el plan de vuelo debe adaptarse a la mejor ubicacion para que las aeronaves no trabajen demasiado lejos ya que la autonomia de las baterias es muy corta y debe optimizarse el vuelo correctamente para aprovechar al maximo la duracion de las baterias. Esto es asi porque si la aeronave se queda sin pesticida o sin bateria demasiado lejos del punto de retorno se utilizará mucha autonomia de la bateria solo para regresar al punto de reinicio del trabajo.

Con relacion al lugar donde se ubicará el equipo reabastecedor debe tenerse en cuenta la cantidad de hectareas que podrá trabajar cada aeronave en cada vuelo. Para esto hay que conocer cuales son las medidas de extension de los lotes y el espaciado entre cada pasada de la aeronave. Por ejemplo, supongamos que el lote tiene una extension este-oeste de 700 metros y el dron trabaja con un interlineado de 8 metros. De esta manera el dron cubrirá 0,56 hectareas en cada pasada. Supongamos que la extension norte-sur del mismo lote es de 1500 metros. En este caso el dron cubrirá 1,2 hectareas en cada pasada. La bateria del dron tiene una duracion de entre 8 a 9 minutos de trabajo. El caudal de aplicacion suele ser de 8 a 10 litros de caldo (mezcla de productos y agua) por hectarea. El tanque de la aeronave tiene capacidad para 40 litros de caldo (en un Agras T50). Todos estos factores son los que deben tenerse en cuenta para planificar la aplicacion de la forma mas eficiente posible. Cuando es posible ubicarse en cualquiera de los extremos del lote la eleccion del lugar para ubicar el equipo reabastecedor y la direccion en la que trabajará la eronave no es tan complicada ya que se elegirá la direccion que permita aprovechar al maximo la duracion de la bateria del mismo. El problema es cuando no tenemos muchos lugares disponibles para ubicar y mover el equipo reabastecedor. En estos casos muchas veces hay que planificar la aplicacion en un sentido que no permite aprovechar al maximo la aplicacion del dron. Si por ejemplo solo podemos realizar la aplicacion en direccion norte-sur (1500 m) tendremos que hacer volver la aeronave luego de dos pasadas ya que no logrará dar una vuelta mas por la limitacion de la autonomia de la bateria y solo podriamos hacer 2,4 hectareas por vuelo (19,2 litros aplicando 8 litros por hectarea o 24 litros usando 10 litros por hectarea). Si pudieramos ubicarnos en el lado del lote que permita planificar la direccion de vuelo de este-oeste (700 metros) podriamos realizar 6 vueltas completas y estariamos aplicando 26,88 litros (a 8 litros por ha) o 33,6 litros (a 10 litros por ha).

En definitiva hay que tener en cuenta todos estos factores para poder lograr un trabajo lo mas eficiente posible en cuanto a los tiempos para finalizar el trabajo lo mas rapido posible.

Otro factor que debe considerarse al momento de planificar el trabajo es la direccion del viento del dia en el que se realizará la aplicacion ya que lo ideal es planificar la direccion del vuelo de modo transversal a la direccion del viento ya que esto mejora la cobertura de aplicacion.

Muchos otros factores se modifican durante la aplicacion dependiendo de las condiciones climaticas en el momento. Estos factores como la altura de trabajo o el tamaño de gota (en micrones) permiten contrarestar los efectos negativos de un viento demasiado fuerte.

Si toda esta planificacion pudiera implementarse en la app seria de mucha ayuda para el usuario.

**Ejemplo de planificacion:** me dirigí hacia la ubicacion de un campo donde habia que realizar una pulverizacion liquida en tres lotes de trigo. El ingeniero agronomo me proporcionó la ubicacion de cada lote y los productos que deseaba utilizar con las dosis correspondientes. Al llegar al campo realizo la medicion del primero de los lotes.

Lote 1: este lote tenia una extension este-oeste de 640 metros. La extension norte-sur era de 700 metros en un extremo y de 1100 metros en el otro extremo por lo que se deduce claramente que la forma del lote es algo uniforme. Anoto las siguientes observaciones: del lado norte del lote hay una calle interna del campo, del lado sur una calle publica, del lado oeste está el lote 2 y al lado este lo defino como la zona apta para aterrizajes y reabastecimiento ya que en este lado hay un lote sin ningun cultivo sembrado.

Lote 2: en este lote mido una extension este-oeste de 580 metros y una extension norte-sur de 700 metros en el extremo mas largo (pegado al lote 1) y de 450 en el extremo mas corto. Anoto que del lado norte está la misma calle interna que pasa del lado norte del lote 1, del lado sur la misma calle publica, del lado este está el lote 1 y el lado oeste es la zona de aterrizajes y reabastecimiento.

Lote 3: este lote tiene una forma cuadrada perfecta por lo que sus lados miden todos 600 metros. Del lado norte hay una calle publica, del lado sur la misma calle interna que pasa por el lado norte de los lotes 1 y 2 (el lote 2 y 3 están separados por esta calle interna) y en los lados este y oeste no hay cultivos sembrados por lo tanto estos son los lados aptos para los aterrizajes y reabastecimiento. Además anoto la observacion de que en el lado este del lote el terreno está un poco barroso.

A la calle interna y a la calle publica se las anota como posibles zonas de aterrizaje en caso de ser necesario.

Debido a que la suma de las extensiones este-oeste de los lotes 1 y 2 es de 1220 metros, la cantidad de hectareas por pasada será de 0,97 has. La pulverizacion se realizará con un caudal de 10 litros por hectarea. En teoria sería posible cargar 39 litros en el tanque del dron para realizar un vuelo de 3,9 hectareas (2 vueltas completas). El problema es que la autonomia de la bateria del dron no permitiría realizar este vuelo sin tener que volver al punto de retorno para cambiar la bateria ya que 39 litros en el tanque es un volumen bastante pesado y a 10 litros por hectarea es casi imposible pulverizar toda esa cantidad en los 8 o 9 minutos que dura una bateria.

En dos vueltas solo se cubrirían 1,95 has por lo que solo sería necesario cargar 19,5 litros en el tanque del dron. Este vuelo tampoco sería eficiente ya que se desperdicia mucho espacio en el tanque y mucha autonomia de bateria.

Debido a todo esto sería muy poco eficiente trabajar los lotes 1 y 2 juntos. Por lo tanto decido pulverizar los lotes por separado primero ubicandome en el lado este del lote 1 y luego en el lado oeste del lote 2.

Ademas el viento pronosticado para el dia en que se iba a realizar el trabajo corría en direccion norte-sur por lo tanto lo ideal era que los drones trabajaran en direccion este-oeste.

Se que implementar toda esta planificacion en mi app puede ser bastante complicado por lo que si no existe una forma de hacerlo de manera sencilla e intuitiva para el usuario talvez lo mejor sería no implementarlo de ninguna manera en la app.

**INFORME EN PDF**

Otra posible implementacion para la app sería la posibilidad de generar un archivo pdf una vez terminado un trabajo con toda la informacion relacionada al trabajo realizado. Muchas empresas de pulverizacion realizan estos informes de forma “manual” para entregarle al cliente una vez terminado un trabajo.

**DOCUMENTACIÓN DE ARQUITECTURA: APP "AL LOTE"**

**1. Visión General del Proyecto**

"Al Lote" es una aplicación Android nativa, desarrollada en **Kotlin**, diseñada como una herramienta de gestión integral para empresas de aplicación de productos fitosanitarios. El sistema permite un control detallado de clientes, trabajos, productos, recetas de aplicación, gestión de lotes y administración financiera, funcionando como un sistema de gestión de recursos empresariales (ERP) móvil y enfocado en el sector agrícola.

La arquitectura de la aplicación es moderna, robusta y sigue las mejores prácticas recomendadas por Google, lo que garantiza su escalabilidad, mantenibilidad y un rendimiento sólido.

**2. Arquitectura General**

La aplicación está construida sobre una arquitectura limpia y modular, siguiendo el patrón **MVVM (Model-View-ViewModel)**. Esta estructura desacopla claramente las responsabilidades en diferentes capas, facilitando el desarrollo, las pruebas y la colaboración.

Las capas principales son:

* **Capa de UI (Vista):** Construida íntegramente con **Jetpack Compose**, es responsable de mostrar los datos en pantalla y de capturar las interacciones del usuario. Las vistas (Composables) son "tontas", es decir, observan un estado (UiState) y notifican eventos al ViewModel, pero no contienen lógica de negocio.
* **Capa de ViewModel:** Actúa como intermediario entre la UI y la capa de datos. Cada pantalla o grupo de pantallas lógicas tiene su propio ViewModel (ej. JobsViewModel, ClientsViewModel). Su función es preparar y gestionar el estado para la UI, manejar la lógica de la interfaz y solicitar datos a los repositorios. Utiliza StateFlow para exponer el UiState a la UI, siguiendo un patrón de **Flujo de Datos Unidireccional (UDF)**.
* **Capa de Repositorio:** Implementa el patrón de repositorio, actuando como la única fuente de verdad (Single Source of Truth) para los datos de la aplicación. Los repositorios (ej. JobsRepository, ProductsRepository) abstraen el origen de los datos (base de datos local, API remota), permitiendo que el resto de la app no necesite saber de dónde vienen los datos.
* **Capa de Fuentes de Datos (Data Sources):** Es la capa más baja y se encarga del acceso directo a los datos. Se divide en:
  + **Fuente de Datos Local:** Gestionada por la base de datos **Room**, que proporciona persistencia local para toda la información crítica (clientes, trabajos, productos, etc.), permitiendo un funcionamiento robusto incluso sin conexión.
  + **Fuente de Datos Remota:** Gestionada por **Retrofit** para consumir APIs externas que proveen datos complementarios como cotizaciones de divisas (Bluelytics), pronósticos del tiempo (Open-Meteo) y noticias del sector (NewsData.io).

**3. Pila Tecnológica (Tech Stack)**

La aplicación utiliza un conjunto de librerías modernas y consolidadas del ecosistema Android:

* **Lenguaje:** **Kotlin** 100%.
* **Interfaz de Usuario (UI):** **Jetpack Compose**, utilizando **Material 3** para los componentes de diseño, lo que garantiza una UI moderna, reactiva y coherente.
* **Arquitectura:**
  + **Guías de Arquitectura de Android:** MVVM, Flujo de Datos Unidireccional (UDF).
  + **Componentes de Arquitectura de Jetpack:** ViewModel, Lifecycle, Navigation.
* **Inyección de Dependencias:** **Hilt** se utiliza para gestionar las dependencias en toda la aplicación, simplificando la creación y provisión de objetos como ViewModels, Repositories y DAOs. Esto es clave para el desacoplamiento y la facilidad de prueba.
* **Persistencia de Datos (Local):** **Room**, un ORM que facilita la creación y gestión de la base de datos SQLite local. Destaca el uso de **Migrations** para manejar las actualizaciones del esquema de la base de datos de forma segura a medida que la aplicación evoluciona.
* **Conectividad (Red):**
  + **Retrofit:** Para realizar llamadas a las APIs REST de forma declarativa.
  + **Gson:** Para la serialización/deserialización de objetos JSON.
* **Asincronía:** **Coroutines de Kotlin** y **Flow** se utilizan en todas las capas para manejar operaciones en segundo plano de manera eficiente, desde las llamadas a la base de datos y la red hasta la actualización de la UI.
* **Navegación:** **Navigation Compose** gestiona la navegación entre las diferentes pantallas (Composables) de la aplicación de forma centralizada y segura.
* **Carga de Imágenes:** **Coil** para cargar y cachear imágenes de manera eficiente, especialmente visible en la sección de noticias.

**4. Estructura del Proyecto y Flujo de Datos**

El proyecto está organizado por funcionalidades (ui.clients, ui.jobs, ui.recetas), lo que facilita la localización del código y el desarrollo en paralelo.

**Flujo de un dato (ej. Cargar la lista de trabajos):**

1. **UI (JobsScreen.kt):** El Composable JobsScreen se muestra. A través de hiltViewModel(), obtiene una instancia de JobsViewModel.
2. **ViewModel (JobsViewModel):** El ViewModel expone un StateFlow<JobsUiState>. En su inicialización, llama a una función del JobsRepository para obtener el stream de trabajos.
3. **Repositorio (JobsRepository):** Recibe la llamada del ViewModel. El repositorio accede a su fuente de datos, que en este caso es el JobDao.
4. **DAO (JobDao):** Es una interfaz de Room que define las consultas a la base de datos. Ejecuta la query SELECT \* FROM jobs y devuelve un Flow<List<Job>>.
5. **Propagación hacia arriba:** El Flow emitido por Room es recogido por el Repositorio, luego por el ViewModel, que lo transforma en el JobsUiState. Finalmente, la UI de JobsScreen observa los cambios en el StateFlow y se recompone automáticamente para mostrar la lista de trabajos actualizada.

**5. Características Clave y Decisiones de Diseño**

* **Base de Datos Robusta con Migraciones:** El archivo AppDatabase.kt muestra una base de datos muy completa con 16 entidades y **22 versiones de migración**. Esto es un indicador excelente de un proyecto maduro y bien mantenido, donde los cambios en la estructura de datos se manejan de forma profesional sin pérdida de información para el usuario.
* **Offline-First:** La arquitectura, con Room como fuente de verdad principal, está diseñada para que la aplicación sea completamente funcional sin conexión a internet. Los datos se leen y escriben localmente, y las integraciones con APIs (clima, noticias) son características complementarias. Esto es ideal para una app de campo.
* **Inyección de Dependencias bien definida (AppModule.kt):** El módulo de Hilt es el cerebro de la arquitectura. Define claramente cómo se construye cada Repositorio y qué DAOs necesita, proveyendo una única instancia (@Singleton) para toda la app. Esto asegura consistencia y eficiencia.
* **Gestión de Estado Centralizada:** El uso del patrón UiState en cada feature asegura que el estado de la UI sea predecible, inmutable y fácil de depurar.
* **UI Tematizada y Consistente:** Los archivos Theme.kt y Color.kt demuestran un sistema de diseño bien pensado, con una paleta de colores semántica (colores de marca, de estado y neutrales) que se aplica de manera consistente en toda la app a través de MaterialTheme.
* **Manejo de Permisos y Recursos del Dispositivo:** El AndroidManifest.xml y el código de la UI muestran un manejo explícito de permisos (localización, almacenamiento) y el uso de APIs del sistema como FileProvider para compartir archivos de forma segura.

**6. Puntos a Considerar y Posibles Mejoras**

* **Pruebas (Testing):** Aunque existen las dependencias de JUnit, no se han proporcionado archivos de prueba. Para asegurar la robustez a largo plazo, sería recomendable añadir:
  + **Pruebas Unitarias:** Para ViewModels y Repositories, verificando la lógica de negocio y las transformaciones de datos.
  + **Pruebas de Integración:** Para la capa de datos (DAOs y migraciones de Room).
  + **Pruebas de UI:** Con Compose Test para verificar que los Composables reaccionan correctamente a los cambios de estado.
* **Manejo de Errores:** La arquitectura permite un manejo de errores robusto (ej. en MainUiState hay un campo error). Sería bueno estandarizar este patrón en todas las UiState para comunicar de forma consistente al usuario problemas de red, de base de datos o de validación.
* **Sincronización con un Backend:** Actualmente, la aplicación es 100% local. Si en el futuro se necesitara una versión multiusuario o con respaldo en la nube, la capa de Repository está perfectamente diseñada para añadir una nueva fuente de datos remota (un backend propio) y gestionar la lógica de sincronización entre la API y la base de datos Room.

**7. Conclusión**

La aplicación "Al Lote" posee una arquitectura de software excelente, moderna y bien implementada. Las decisiones de diseño, como el uso de MVVM, Jetpack Compose, Hilt y Room, están alineadas con las mejores prácticas de la industria. La estructura modular, el claro desacoplamiento de capas y el manejo cuidadoso de la base de datos la convierten en una base sólida y escalable, preparada para crecer y añadir nuevas funcionalidades con facilidad. Cualquier desarrollador que se incorpore al proyecto encontrará un código organizado, predecible y fácil de entender.

**PANTALLAS PRINCIPALES**

* **DashboardScreen:** Esta pantalla, DashboardScreen, opera como el centro de control principal de la aplicación, proporcionando una visión general rápida y concisa de la información más relevante para el usuario. Su objetivo primordial es ofrecer un "tablero" interactivo que consolide datos clave y permita el acceso rápido a funcionalidades esenciales, mejorando la eficiencia y la toma de decisiones.Al iniciar, la pantalla solicita permisos de ubicación, vitales para la funcionalidad climática. Una vez obtenidos, o en su ausencia, procede a cargar y mostrar el contenido. Incorpora un indicador de carga inicial y soporta la funcionalidad de "pull-to-refresh" para que el usuario pueda actualizar todos los datos fácilmente, asegurando que la información esté siempre al día.

Uno de sus componentes centrales es la sección de clima, que presenta un resumen del tiempo actual, incluyendo temperatura y tipo de clima. Esta tarjeta es expandible, revelando detalles adicionales como la dirección del viento, la humedad y la precipitación diaria, y ofrece un botón para acceder a un pronóstico horario más detallado a través de un diálogo.La pantalla también incluye una sección de "Acceso Rápido" que facilita la navegación directa a módulos clave de la aplicación, como "Trabajos", "Clientes", "Administración", "Productos" y "Checklists", optimizando el flujo de trabajo del usuario. Complementando esto, la sección de "Resumen" muestra estadísticas vitales del negocio en tiempo real, como el número de trabajos pendientes y hectáreas asociadas, la cantidad total de clientes y el saldo financiero general, que se presenta con un indicador visual de positivo o negativo.

Finalmente, una destacada sección de "Noticias del Agro" mantiene al usuario informado sobre novedades relevantes del sector, con la capacidad de cargar más artículos automáticamente a medida que el usuario se desplaza (scroll infinito). Esta sección maneja sus propios estados de carga y errores, proporcionando retroalimentación clara si hay problemas para obtener las noticias. En su conjunto, la pantalla busca ser una herramienta informativa y práctica, diseñada para mantener al usuario al tanto y facilitar sus operaciones diarias.

* **ClientScreen:** Esta pantalla, denominada ClientsScreen, sirve como el hub central para la gestión de clientes dentro de la aplicación. Su objetivo principal es permitir a los usuarios visualizar, buscar, añadir, editar y eliminar la información de sus clientes de una manera intuitiva y eficiente.Al cargar, la pantalla muestra una barra superior con el título "Clientes" y un botón de ayuda que, al ser pulsado, despliega un diálogo informativo con instrucciones detalladas sobre cómo interactuar con las diversas funcionalidades. Debajo de la barra superior, se presenta un campo de texto con un icono de búsqueda que permite al usuario filtrar la lista de clientes ingresando texto; si hay una búsqueda activa, aparece un icono para borrar el texto y limpiar el filtro.La parte central de la pantalla se dedica a mostrar la lista de clientes. Si la aplicación está cargando los datos, se visualiza un indicador de progreso circular. En caso de que no haya clientes registrados o que la búsqueda no arroje resultados, se muestra un mensaje de estado vacío con un icono relevante, guiando al usuario sobre cómo añadir su primer cliente. Cuando hay clientes disponibles, estos se presentan en una lista desplazable, donde cada cliente tiene su propia tarjeta. Cada tarjeta de cliente muestra el nombre completo y, si está disponible, el número de teléfono del cliente. Además, cada tarjeta incluye iconos para editar y eliminar al cliente.La interacción con los clientes es variada: al pulsar sobre una tarjeta de cliente, el usuario es redirigido a una vista de detalles del cliente para acceder a su panel de administración, trabajos y contabilidad. El icono de lápiz permite editar la información del cliente, abriendo un diálogo específico (ClientDialog) que precarga los datos del cliente seleccionado para su modificación. El icono de papelera, por su parte, activa un diálogo de confirmación antes de eliminar permanentemente al cliente de la base de datos. Finalmente, la pantalla se integra con una acción flotante (Floating Action Button) definida en otro lugar de la aplicación, que al ser pulsado, permite al usuario añadir un nuevo cliente abriendo el mismo diálogo de cliente, pero con campos vacíos. En resumen, ClientsScreen está diseñada para ofrecer una experiencia completa y fluida en la administración del directorio de clientes.
* **JobsScreen:** Esta pantalla, denominada JobsScreen, es el centro de gestión para todos los trabajos dentro de la aplicación, diseñada para proporcionar una visión clara y control sobre las operaciones. Su objetivo principal es permitir a los usuarios visualizar el estado de sus trabajos, filtrarlos, y realizar acciones clave como añadir, editar, eliminar o gestionar aspectos específicos de cada uno.Al cargar, la pantalla presenta una barra superior centrada con el título "Gestión de Trabajos", que incluye un icono de ayuda. Al pulsar este icono, se despliega un diálogo (HelpDialog) que ofrece una explicación concisa sobre el funcionamiento de la pantalla y sus características principales. El contenido principal se muestra a continuación, comenzando con un indicador de progreso circular si los datos aún se están cargando. Si no hay trabajos registrados en el sistema, se presenta un estado vacío (EmptyState) que guía al usuario para crear su primer trabajo utilizando el botón flotante inferior, que aunque no está definido en este archivo, se infiere su existencia en la arquitectura general de la aplicación.Cuando hay trabajos disponibles, la pantalla organiza la información de manera estructurada. Inicialmente, se muestra un encabezado interactivo (JobsHeader) que resume el total de trabajos registrados y las hectáreas pendientes. Este encabezado también incluye un botón "Filtrar" que, al ser pulsado, abre un diálogo de filtros (FilterDialog). Este diálogo permite al usuario refinar la lista de trabajos por diversos criterios, como cliente, estado (Pendiente/Finalizado), tipo de aplicación, estado de facturación (No Facturado, Facturado, Pagado), y un rango de fechas "Desde" y "Hasta", utilizando selectores de fecha intuitivos.Los trabajos se presentan en dos secciones diferenciadas: "Pendientes" y "Finalizados", cada una con un título y un contador que indica la cantidad de trabajos en esa categoría. Dentro de cada sección, los trabajos se listan individualmente a través del componente JobListItem. Cada elemento de la lista muestra el nombre del cliente, la superficie en hectáreas, el tipo de aplicación (con un icono y color distintivo), y una breve descripción si está disponible. Al pulsar sobre un trabajo en la lista, se desencadena una acción para ver sus detalles completos, mientras que al pulsar el icono de tres puntos ("Opciones") en cada ítem, se abre un diálogo (JobOptionsDialog) que ofrece un conjunto de acciones rápidas. Estas acciones incluyen marcar el trabajo como finalizado (lo que abre un selector de fecha para registrar la finalización), editar el trabajo (abriendo un formulario (JobDialogCompose) precargado con los datos del trabajo para su modificación), gestionar su facturación (mediante otro diálogo que permite cambiar el estado de facturación), ver las recetas asociadas al trabajo, o eliminarlo por completo. La adición de nuevos trabajos se realiza a través del mismo JobDialogCompose, pero con campos vacíos. Finalmente, la pantalla proporciona retroalimentación al usuario mediante mensajes Toast para confirmar acciones como el guardado, actualización o eliminación de trabajos.
* **ClientAdminScreen:** Esta pantalla, denominada ClientAdministracionScreen, actúa como un panel de control específico para la gestión de un cliente individual dentro de la aplicación. Su objetivo principal es ofrecer al usuario una vista detallada de un cliente seleccionado y proporcionar accesos directos a las funcionalidades clave relacionadas con ese cliente, como la gestión de sus trabajos y su contabilidad.Al iniciar, la pantalla está diseñada para recibir un objeto Client y, si este objeto aún no está disponible (por ejemplo, mientras se carga desde una base de datos o API), muestra un indicador de progreso circular (CircularProgressIndicator) en el centro de la pantalla, informando al usuario que la información está siendo cargada. La barra superior (TopAppBar) de la pantalla se actualiza dinámicamente para mostrar el nombre completo del cliente una vez que los datos han sido cargados, o el texto "Cargando..." si aún no lo están.Una vez que la información del cliente está disponible, la interfaz principal se organiza en una columna centralizada. En la parte superior, se muestra una imagen decorativa que ocupa todo el ancho de la pantalla y tiene una altura fija, utilizando coil.compose.rememberAsyncImagePainter para cargarla desde una URL predefinida. Debajo de esta imagen, y tras un espaciador para mejorar la legibilidad, se presentan los detalles del cliente. Esta sección muestra información vital como el nombre, apellido y número de teléfono del cliente, asegurando que solo se muestren los campos que contienen datos válidos y no están vacíos.Posteriormente, se incluye un texto informativo que guía al usuario a seleccionar una opción para administrar al cliente. A continuación, la pantalla ofrece dos botones principales que sirven como puntos de navegación clave: el botón "Trabajos" y el botón "Contabilidad". Al pulsar el botón "Trabajos", la aplicación navega a la sección donde se pueden visualizar y gestionar todos los trabajos asociados específicamente a este cliente. De manera similar, al pulsar el botón "Contabilidad", el usuario es dirigido a la sección financiera que corresponde a este cliente, permitiendo el acceso a sus registros contables. Ambas acciones de navegación se realizan pasando el identificador único del cliente (ID), lo que garantiza que las pantallas subsiguientes muestren información relevante solo para el cliente actual. En resumen, ClientAdministracionScreen es una interfaz dedicada que centraliza la administración y el acceso a las operaciones específicas de cada cliente.
* **ClientJobsScreen:** Esta pantalla, denominada ClientJobsScreen, está diseñada como el punto central para la visualización y gestión de todos los trabajos asociados a un cliente específico dentro de la aplicación. Su objetivo principal es ofrecer al usuario una vista detallada de los trabajos de un cliente, permitiendo filtrarlos, realizar acciones específicas sobre ellos y facilitar la adición de nuevos trabajos.Al cargar, la pantalla presenta una barra superior que muestra el nombre completo del cliente al que pertenecen los trabajos, ofreciendo una clara contextualización. En esta misma barra superior, se encuentran dos iconos de acción: uno para acceder a un diálogo de ayuda (HelpDialog) que explica el uso de la pantalla, y otro para navegar directamente a la sección de contabilidad de este cliente, optimizando el acceso a información financiera relevante.El cuerpo principal de la pantalla gestiona la carga de datos, mostrando un indicador de progreso circular (CircularProgressIndicator) mientras los trabajos son recuperados. Una vez cargados, se ofrece al usuario un botón prominente para "Mostrar Filtros", que al ser pulsado despliega el diálogo ClientJobsFilterDialog. Este diálogo es una herramienta fundamental, permitiendo a los usuarios refinar la lista de trabajos por diversos criterios como el estado (pendiente, finalizado), el tipo de aplicación (líquida, sólida, mixta, varias), el estado de facturación (no facturado, facturado, pagado) y un rango de fechas de inicio y fin, facilitando la búsqueda y organización de la información.La lista de trabajos se presenta en una columna desplazable (LazyColumn), donde cada trabajo se muestra como un ítem de tarjeta (ClientJobListItem). Cada tarjeta proporciona un resumen visual del trabajo, incluyendo su descripción (o un mensaje predeterminado si no la tiene), la superficie en hectáreas, y el estado actual del trabajo (con iconos y colores distintivos para "Pendiente" o "Finalizado"). Al pulsar sobre cualquier tarjeta de trabajo, el usuario es redirigido a una vista más detallada de ese trabajo específico. Adicionalmente, cada tarjeta incluye un icono de "más opciones" (MoreVert), que al ser pulsado, abre el JobActionsDialog. Este diálogo contextual ofrece una serie de acciones rápidas sobre el trabajo seleccionado, como editar sus detalles, marcarlo como finalizado (con un selector de fecha para registrar la fecha de fin), actualizar su estado de facturación mediante un sub-diálogo, ver las recetas asociadas a ese trabajo, o eliminarlo de forma permanente.La funcionalidad para añadir nuevos trabajos se integra con un botón de acción flotante (FAB), que está configurado para abrir el JobDialogCompose. Este mismo diálogo se utiliza tanto para la creación de nuevos trabajos (con campos vacíos) como para la edición de trabajos existentes (precargando los datos del trabajo seleccionado). Tras cualquier acción de guardado o actualización de un trabajo, la aplicación proporciona retroalimentación visual al usuario a través de mensajes Toast, confirmando que la operación se ha realizado con éxito. En resumen, ClientJobsScreen es una interfaz completa y dinámica diseñada para centralizar y simplificar la gestión de los trabajos de cada cliente.
* **JobDetailScreen:** Esta pantalla, denominada JobDetailScreen, es el centro neurálgico para la visualización y gestión exhaustiva de un trabajo específico dentro de la aplicación. Su propósito fundamental es ofrecer al usuario una visión completa y detallada de un trabajo seleccionado, junto con accesos directos a todas las funcionalidades y herramientas relevantes asociadas a dicho trabajo.Al iniciar, la pantalla muestra una barra superior que presenta la descripción del trabajo como título, y un icono de ayuda que, al ser pulsado, despliega un diálogo informativo (HelpDialog) explicando las distintas secciones y funcionalidades disponibles. Durante la carga de la información del trabajo, se visualiza un indicador de progreso, y en caso de que el trabajo no se encuentre, se informa al usuario con un mensaje de error.Una vez cargados los datos, la pantalla se organiza en varias secciones clave. Primero, se muestra un JobSummaryCard que ofrece un resumen conciso del trabajo, incluyendo el nombre del cliente, la superficie en hectáreas y el estado actual del trabajo (pendiente, en progreso, completado), utilizando un diseño visualmente atractivo con gradientes. A continuación, el JobDetailsCard profundiza en los atributos del trabajo, detallando la fecha de creación, las fechas reales de inicio y finalización (si están establecidas), el estado de facturación y cualquier nota relevante, presentando esta información de manera clara y organizada con iconos y colores para una fácil comprensión.Un componente crucial es el JobActionMenu, que funciona como un panel de control interactivo. Este menú presenta una cuadrícula de tarjetas de acción, cada una representando una funcionalidad específica vinculada al trabajo, como "Costos" para la facturación, "Parámetros", "Recetas" para la creación de mezclas, "Ubicación" para ver o establecer las coordenadas geográficas del trabajo, "Imágenes" para gestionar fotos, y "Lotes" para dividir y registrar superficies tratadas. Al pulsar cualquiera de estas tarjetas, el usuario es redirigido a la pantalla correspondiente con el contexto de este trabajo específico.La gestión de la ubicación es una característica importante; si la ubicación del trabajo no está definida, se ofrece un LocationPickerDialog que permite al usuario fijarla en un mapa. Si ya existe, se abre un LocationViewerDialog para visualizarla, con la opción de editarla si es necesario.Finalmente, la pantalla integra una sección de ForecastSection que muestra un pronóstico climático semanal relevante para la ubicación del trabajo. Cada día del pronóstico se presenta en una ForecastCard que incluye temperatura máxima y mínima, tipo de clima y datos de precipitación y viento. Al pulsar sobre una tarjeta de pronóstico diario, se activa un HourlyForecastDialog que despliega un pronóstico horario detallado para ese día, incluyendo una barra visual (EnhancedTemperatureBar) que indica la temperatura actual dentro del rango diario, junto con detalles de precipitación, viento y humedad por hora. En su totalidad, JobDetailScreen está diseñada para ser un centro de información y acción completo, proporcionando al usuario todas las herramientas necesarias para gestionar un trabajo de principio a fin.
* **AdministracionScreen:** Esta pantalla, denominada AdministracionScreen, está concebida como la interfaz principal para la gestión y cálculo de los aspectos económicos de un trabajo específico dentro de la aplicación. Su objetivo primordial es permitir al usuario introducir datos clave como la superficie trabajada y el costo por hectárea, aplicar impuestos si es necesario, realizar los cálculos financieros correspondientes y, finalmente, guardar esta información.Al iniciar, la pantalla presenta una barra superior animada (AnimatedTopBar) que muestra el título "Administración del Trabajo" y un icono de ayuda. Al pulsar este icono, se despliega un diálogo de ayuda animado (AnimatedHelpDialog) que proporciona una guía visual y textual sobre cada uno de los campos de entrada y botones de acción presentes en la pantalla, explicando su propósito y funcionamiento. La pantalla también incorpora un estado de carga (isLoading), durante el cual se muestra una pantalla de carga animada (AnimatedLoadingScreen) con un indicador de progreso giratorio.El cuerpo principal de la pantalla está estructurado en varias secciones interactivas y visualmente atractivas. La primera es la "Hero Section" animada (AnimatedHeroSection), que destaca la descripción del trabajo actual. En esta sección, el usuario puede ingresar o ajustar la "Superficie a Facturar" en hectáreas mediante un campo de texto flotante, que se integra visualmente con animaciones de escala.A continuación, se encuentra el "Panel de Cálculo" interactivo (AnimatedCalculationPanel). Este panel es donde el usuario introduce el "Precio por Hectárea" en un campo de texto con formato numérico. Una característica clave es el interruptor animado "Aplicar IVA", que permite al usuario decidir si se debe incluir el impuesto del 10.5% en los cálculos finales. Después de introducir los valores, el usuario puede pulsar el botón "Calcular" para ver los resultados de forma inmediata. Un segundo botón, "Guardar", permite almacenar estos cálculos en el sistema, confirmando la acción con un mensaje Toast flotante.Finalmente, la pantalla presenta la "Sección de Resultados" animada (AnimatedResultsSection), que se hace visible (AnimatedVisibility) solo después de que se ha realizado un cálculo. Esta sección muestra de forma clara y organizada los resultados financieros, incluyendo el número de "Hectáreas" ingresadas, el "Precio/Ha", el "Subtotal" (el costo sin IVA) y el "Total + IVA" (el costo final con el impuesto aplicado). Cada uno de estos resultados se presenta en una tarjeta animada individual (AnimatedResultCard), destacando los valores con iconos y colores distintivos, y mostrando los montos de dinero formateados según la configuración de moneda de la aplicación. En su conjunto, AdministracionScreen ofrece una experiencia de usuario dinámica y visualmente rica para la gestión financiera de los trabajos, combinando la entrada de datos, el cálculo y la visualización de resultados con transiciones y animaciones fluidas.
* **ParametrosScreen:** Esta pantalla, denominada ParametrosScreen, tiene como objetivo principal permitir al usuario visualizar y gestionar los parámetros específicos asociados a un trabajo dentro de la aplicación. Su funcionalidad se adapta dinámicamente según el tipo de aplicación del trabajo (sólida o líquida), mostrando los campos relevantes para cada caso.Al cargar, la pantalla determina si ya existen parámetros guardados para el trabajo actual. Si no hay parámetros (es decir, la dosis es nula), la pantalla entra automáticamente en un modo de edición para que el usuario pueda configurarlos por primera vez. De lo contrario, se muestra una vista resumida de los parámetros existentes. Durante la carga de datos, se visualiza un indicador de progreso, y si el trabajo no se encuentra, se notifica al usuario con un mensaje de error.En el modo de visualización, la pantalla presenta una tarjeta (ParametrosSummaryCard) que muestra de forma concisa los parámetros guardados, como la dosis (con la unidad adecuada, Kg/ha o L/ha), interlineado, velocidad y altura. Dependiendo de si la aplicación es sólida, también se mostrarán el disco utilizado y las revoluciones; si es líquida, se indicará el tamaño de gota. Un texto guía al usuario, informándole que puede mantener presionado cualquier parte del resumen para activar el modo de edición. Un botón flotante de edición (FloatingActionButton) también está disponible para alternar a este modo.Cuando la pantalla entra en modo de edición (ParametrosEditForm), se presentan campos de texto interactivos (OutlinedTextField) para cada parámetro. El usuario puede introducir o modificar la dosis, interlineado, velocidad y altura. Si el trabajo es de aplicación sólida, aparecerán campos adicionales para seleccionar el "Disco Utilizado" mediante un desplegable y las "Revoluciones" (RPM). En contraste, si es de aplicación líquida, se habilitará un campo para el "Tamaño de Gota" (μm). Todos los campos de entrada numérica están configurados para aceptar solo números. Al finalizar la edición, un botón flotante extendido "Guardar Parámetros" permite al usuario confirmar los cambios, que luego se persisten y se le notifica mediante un mensaje breve, regresando a la pantalla anterior. La transición entre el modo de visualización y el modo de edición se realiza de manera fluida mediante animaciones de aparición y desaparición, proporcionando una experiencia de usuario cohesiva. En resumen, ParametrosScreen es una interfaz dedicada a la configuración detallada de los parámetros operativos de cada trabajo, garantizando la precisión y adaptabilidad según la naturaleza de la tarea.
* **RecetasScreen:**  Esta pantalla, denominada RecetasScreen, es el componente central para la creación y gestión de recetas de aplicación dentro de un trabajo específico en la aplicación. Su objetivo principal es permitir a los usuarios definir los parámetros de aplicación, seleccionar los productos necesarios con sus dosis correspondientes, calcular las cantidades totales requeridas y visualizar un resumen detallado de la receta resultante.Al iniciar, la pantalla verifica el estado de carga y muestra un indicador de progreso circular si la información aún se está recuperando. La barra superior contextualiza la pantalla mostrando el nombre del cliente asociado al trabajo y ofrece un acceso rápido a un diálogo de ayuda mejorado (EnhancedHelpDialog), que explica paso a paso cómo utilizar la funcionalidad de recetas y lotes con un diseño visualmente atractivo y secciones claras.La interfaz se divide en varias secciones clave. La primera es la "Sección de Entradas Mejorada" (EnhancedInputSection), donde el usuario introduce los parámetros fundamentales para la aplicación. Estos incluyen las hectáreas a tratar, el caudal de aplicación por hectárea (específico para aplicaciones líquidas) y la capacidad del tanque de la pulverizadora por carga, que se adapta para mostrar "Caldo por tachada (L)" o "Kgs por tachada" si la aplicación es sólida. Cada campo de entrada (EnhancedTextField) incluye validación en tiempo real y mensajes de error si los valores son incorrectos, todo presentado dentro de una tarjeta con un degradado de color y un encabezado distintivo.A continuación, se encuentran los "Botones de Acción Mejorados" (EnhancedActionButtons). Aquí, el usuario puede pulsar "Agregar Producto" para abrir un diálogo (SelectProductDialog) que permite buscar y seleccionar productos de una lista. Una vez seleccionado un producto, se abre otro diálogo (DoseInputDialog) para especificar la dosis por hectárea, ya sea en litros o gramos, dependiendo del tipo de producto. El botón "Calcular" es fundamental, ya que procesa todos los parámetros y productos añadidos para generar el resumen de la receta, estando habilitado solo si todos los datos obligatorios son válidos y hay productos en la receta. Además, hay un botón "Gestionar Lotes" que navega a la pantalla de lotes, permitiendo al usuario dividir el trabajo total en secciones más pequeñas para un seguimiento detallado.La "Sección de Productos Mejorada" (EnhancedProductsSection) muestra de manera clara y organizada todos los productos que han sido añadidos a la receta. Cada producto aparece como un ítem individual (EnhancedProductoRecetaItemView), detallando su nombre comercial, la dosis por hectárea y la cantidad total requerida, con una barra lateral codificada por colores que indica su banda toxicológica. Cada ítem también incluye un botón para eliminar el producto de la receta.

Finalmente, la "Sección de Resumen Mejorada" (EnhancedSummarySection) presenta el resultado de los cálculos. Esta sección muestra un texto conciso que resume la receta, y si los parámetros o productos han sido modificados después del último cálculo, una barra de color y un icono de advertencia (Warning) indican que el resumen está "sucio" y necesita ser recalculado para reflejar los cambios. La tarjeta de resumen permite una pulsación larga para acceder a un menú de acciones relacionadas con la receta (aunque este menú es actualmente un marcador de posición). En resumen, RecetasScreen es una herramienta robusta y visualmente enriquecida que guía al usuario a través del proceso de creación de recetas de aplicación precisas y eficientes para sus trabajos.

* **ImagesJobScreen:** Esta pantalla, denominada ImagesJobScreen, está dedicada a la gestión visual de los trabajos dentro de la aplicación, permitiendo al usuario adjuntar, visualizar y eliminar imágenes asociadas a un trabajo específico. Su objetivo primordial es proporcionar un registro fotográfico del progreso o los detalles de cada tarea, mejorando la documentación y el seguimiento.Al iniciar, la pantalla presenta una barra superior con el título "Imágenes del Trabajo". El flujo principal comienza con un botón prominente que invita al usuario a "Seleccionar Imágenes". Al pulsarlo, se activa un lanzador de actividad (rememberLauncherForActivityResult) que permite seleccionar múltiples imágenes de la galería del dispositivo. Una vez que las imágenes son elegidas, la aplicación solicita y toma permisos persistentes para acceder a sus URIs, asegurando que las imágenes puedan ser visualizadas más tarde, incluso si la aplicación se cierra. Tras la selección, se muestra un mensaje Toast informando al usuario sobre la cantidad de imágenes añadidas o si no se seleccionó ninguna.El cuerpo principal de la pantalla maneja diferentes estados. Si la aplicación está cargando las imágenes (por ejemplo, al inicio o después de una acción), se muestra un indicador de progreso circular (CircularProgressIndicator). Si no hay imágenes asociadas al trabajo, la pantalla presenta un estado vacío (EmptyState), con un icono de imagen y un texto que sugiere al usuario añadir fotos para un registro visual.Cuando hay imágenes disponibles, estas se muestran en una lista desplazable verticalmente (LazyColumn). Cada imagen se representa mediante el componente ImageRow, que consiste en una tarjeta interactiva. Dentro de cada tarjeta, se incluye una pequeña vista previa de la imagen, cargada eficientemente con la librería Coil, que ocupa un espacio de 80dp. Al lado de la vista previa, se muestra un texto genérico "Imagen" (aunque se podría mejorar para mostrar el nombre del archivo). Un aspecto clave de la funcionalidad es que al pulsar en cualquier parte de la tarjeta de la imagen, la aplicación intenta abrir la imagen en una aplicación externa del sistema, como un visor de fotos, utilizando una Intent con permisos de lectura. Si no se encuentra una aplicación compatible, se informa al usuario mediante un Toast. Finalmente, cada ImageRow incluye un botón con un icono de papelera (Delete), que al ser pulsado, permite al usuario eliminar la imagen específica del trabajo, ofreciendo un control total sobre el contenido visual. En resumen, ImagesJobScreen es una interfaz intuitiva para la documentación fotográfica de los trabajos, facilitando la adición, visualización y gestión de las imágenes pertinentes.
* **GestionLotesScreen:** La pantalla GestionLotesScreen actúa como una herramienta de microgestión para un trabajo específico, permitiendo al usuario dividir la superficie total planificada en sub-secciones manejables denominadas 'lotes'. Su objetivo principal es facilitar el seguimiento preciso del trabajo de campo y la gestión de insumos. El funcionamiento se basa en un flujo de trabajo claro: el usuario primero crea los lotes a través de un diálogo, especificando un nombre y las hectáreas planificadas para cada uno, con una validación que impide superar la superficie total del trabajo.

Una vez creados los lotes, la pantalla permite registrar la ejecución real del trabajo. Mediante una pulsación larga sobre un lote, el usuario puede acceder a la opción 'Registrar Trabajo Realizado' para introducir la superficie efectivamente tratada, que puede diferir de la planificada. Esta diferenciación entre lo planificado y lo real es el eje central de la pantalla. Cuando al menos un lote tiene su trabajo real registrado, se habilita la función clave: un botón para 'Generar Resumen de Sobrantes'. Al pulsarlo, el sistema calcula la diferencia entre los insumos preparados para la superficie planificada y los realmente utilizados en la superficie trabajada, presentando un informe detallado del producto sobrante.

Adicionalmente, la pantalla ofrece otras interacciones: una pulsación corta sobre un lote muestra un resumen de la receta de productos específica para su superficie planificada, y la pulsación larga también da acceso a funciones para editar, eliminar o asignar coordenadas GPS a cada lote.

* **ClientContabilidadScreen:** La pantalla ClientContabilidadScreen está diseñada para ofrecer una vista detallada y específica de la cuenta corriente de un único cliente. Su objetivo principal es presentar un historial financiero completo que refleje todos los débitos (costos de trabajos) y créditos (pagos realizados) de ese cliente en particular, permitiendo tanto al administrador como al cliente tener un entendimiento claro de su estado de cuenta.

El funcionamiento de la pantalla se articula en torno a una tarjeta de balance (BalanceCard) que ocupa un lugar central, mostrando de manera inmediata si el cliente tiene un saldo a favor (acreedor) o una deuda pendiente (deudor). Justo debajo, una lista cronológica presenta cada movimiento contable, mostrando la descripción, la fecha y las columnas de "debe" y "haber" correspondientes. Esta lista puede ser acotada en el tiempo mediante un botón de filtro que permite al usuario seleccionar un rango de fechas específico para su consulta.

La interacción con los movimientos es completa: el usuario puede añadir nuevas transacciones, como pagos o notas de crédito, a través del botón flotante de acción. Una pulsación larga sobre un movimiento existente despliega un menú contextual para editarlo o eliminarlo, mientras que una pulsación corta permite ver detalles adicionales o abrir documentos adjuntos si los hubiera. Una lógica de negocio importante se activa al intentar eliminar un movimiento que ya ha sido aprobado en la contabilidad general; en este caso, el sistema muestra un diálogo de advertencia que solicita confirmación para asegurar que la eliminación se sincronice en ambas secciones, manteniendo la integridad de los datos financieros globales.

* **AdministracionGeneralScreen:** La pantalla de AdministracionGeneralScreen funciona como el centro de control financiero de la aplicación, ofreciendo una visión global de todos los movimientos contables de la empresa. Su objetivo principal es consolidar ingresos y egresos, tanto los que provienen de trabajos específicos con clientes como aquellos que son gastos o ingresos generales del negocio, para presentar un estado de cuenta unificado y claro.

El funcionamiento de la pantalla se centra en un listado cronológico de transacciones. En la parte superior, una tarjeta de balance (BalanceCard) muestra de forma prominente el saldo total del período de tiempo seleccionado, permitiendo al usuario conocer rápidamente la salud financiera de su operación. La interfaz está dominada por una lista de movimientos (LazyColumn) que se agrupa por mes, facilitando la navegación y la consulta histórica. Cada elemento de la lista detalla la descripción del movimiento, el cliente asociado si corresponde, la fecha, y las columnas de egresos e ingresos.

La interacción del usuario es rica y funcional. La pantalla incluye un potente sistema de filtrado por fechas accesible desde la barra superior, que permite al usuario visualizar los datos por día, semana, mes o un rango personalizado. Para la gestión de transacciones, una pulsación larga sobre un movimiento despliega un menú de opciones para editarlo o eliminarlo. Una pulsación corta, en cambio, permite ver los detalles del movimiento o, si tiene documentos adjuntos, navegar a un visor dedicado.

Una característica clave de esta pantalla es el flujo de aprobación de pagos. Cuando un cliente registra un pago desde su propia sección, este no se incorpora inmediatamente a la contabilidad general. En su lugar, aparece una notificación en la parte superior de esta pantalla, indicando que hay movimientos pendientes de revisión. El administrador debe pulsar esta notificación para acceder a un diálogo donde puede aprobar o rechazar dichos pagos, asegurando un control total sobre los ingresos que se acreditan a la empresa. Finalmente, el botón flotante de acción permite registrar nuevos movimientos generales que no están vinculados a un trabajo específico, como la compra de insumos o el pago de servicios.

* **ProductsScreen:** La pantalla ProductsScreen funciona como un catálogo centralizado de insumos agrícolas, y su objetivo principal es permitir al usuario gestionar todos los productos que utiliza en sus trabajos. Esta sección es fundamental en el flujo de la aplicación, ya que los productos registrados aquí son los que estarán disponibles posteriormente para la creación de recetas de aplicación en la RecetasScreen.

La interfaz se organiza de manera clara mediante dos pestañas principales que categorizan los insumos por su tipo de aplicación: "Pulverización" y "Esparcido". La categoría de pulverización está destinada a productos que se mezclan en un caldo líquido, como herbicidas o insecticidas, y requiere que se les asigne una formulación para determinar el orden de mezcla. Por otro lado, la sección de esparcido agrupa productos que se aplican directamente en estado sólido, como semillas o fertilizantes granulados.

Para añadir un nuevo producto, el usuario utiliza el botón flotante de acción, que presenta un formulario adaptado a la categoría seleccionada. Una vez que los productos están listados, la pantalla ofrece funcionalidades de gestión completas. Un campo de búsqueda permite filtrar la lista por nombre para una localización rápida. La interacción con cada producto es intuitiva: una pulsación corta sobre un elemento navega a su pantalla de detalle para una edición exhaustiva, mientras que una pulsación larga inicia el proceso de eliminación a través de un diálogo de confirmación. A nivel visual, cada producto en la lista no solo muestra su nombre comercial y un dato clave, sino que también incluye una barra de color vertical que representa su banda toxicológica, ofreciendo una identificación inmediata de su peligrosidad.

En esencia, esta pantalla actúa como el inventario maestro de insumos, y su correcta gestión es un prerrequisito indispensable para poder utilizar la funcionalidad de creación de recetas, una de las características centrales del sistema.

* **ProductDetailScreen:** Esta pantalla, denominada ProductDetailScreen, está diseñada para permitir al usuario visualizar y modificar los detalles de un producto específico dentro de la aplicación. Su objetivo primordial es ofrecer una interfaz clara y funcional para la gestión individual de los productos, permitiendo actualizar sus atributos y adaptarse a diferentes tipos de aplicación.Al cargar, la pantalla maneja varios estados. Si la información del producto aún se está recuperando o procesando, se muestra un indicador de progreso circular (CircularProgressIndicator) en el centro de la pantalla. En el caso excepcional de que el producto no sea encontrado o sea nulo, se presenta un mensaje informativo indicando esta situación.Una vez que el producto está disponible, la pantalla se organiza en una columna desplazable que presenta los campos editables del producto. El título principal de la pantalla es "Detalle del Producto". Los campos base, como el "Nombre Comercial", están siempre visibles y se actualizan en tiempo real a medida que el usuario escribe. Una característica clave de esta pantalla es su adaptabilidad: si el tipo de aplicación del producto es "Pulverización", se revelan campos adicionales y específicos para este contexto. Estos incluyen el "Principio Activo" (otro campo de texto libre), y tres selectores desplegables (ExposedDropdownMenuBox) para elegir el "Tipo" de producto (como Herbicida, Insecticida, Fungicida, etc., de una lista predefinida), la "Formulación" (seleccionable de una lista proporcionada por el estado de la UI), y la "Banda Toxicológica" (elegida entre opciones como Ia, Ib, II, III, IV). Cada uno de estos desplegables permite al usuario seleccionar una opción de forma intuitiva, cerrándose automáticamente una vez que se ha hecho una elección.Finalmente, en la parte inferior de la pantalla, se encuentra un botón prominente "Guardar Cambios". Al pulsar este botón, la aplicación recopila los valores actuales de todos los campos modificados, crea una versión actualizada del objeto Product y la envía a través de una función de callback (onUpdateProduct) para su persistencia en la capa de datos. Para proporcionar retroalimentación inmediata al usuario, se muestra un mensaje breve (Toast) confirmando que el producto ha sido actualizado exitosamente. En resumen, ProductDetailScreen es una interfaz completa y adaptable para la edición detallada de productos, asegurando que la información esté siempre correcta y relevante para el tipo de aplicación.
* **SettingsScreen:** Esta pantalla, denominada SettingsScreen (Configuración), es el área designada para que el usuario personalice diversos aspectos de la aplicación, adaptándola a sus preferencias y necesidades operativas. Su objetivo principal es ofrecer un control exhaustivo sobre configuraciones monetarias, precios de servicios y otras opciones informativas.La interfaz general se organiza bajo una barra superior que muestra el título "Configuración". El contenido principal de la pantalla es completamente desplazable, garantizando que todas las opciones sean accesibles. Se inicia con un encabezado distintivo (SettingsHeader) que da la bienvenida al usuario al área de configuración.A continuación, la pantalla se divide en secciones temáticas, cada una con un título e icono representativo. La sección de "Configuración Monetaria" es fundamental para adaptar la aplicación a las realidades financieras del usuario. Aquí se encuentra una opción para seleccionar la "Moneda de Visualización", permitiendo al usuario elegir entre USD y ARS a través de un diálogo específico. Si la moneda seleccionada es ARS, se revelan dos opciones adicionales: la "Tasa de Cambio" entre USD y ARS, que puede ser editada manualmente mediante un diálogo numérico, y un botón para "Actualizar Tasa Automáticamente", que al ser pulsado, inicia una operación para obtener la cotización del dólar blue desde una fuente externa, mostrando un indicador de carga durante este proceso.La sección de "Precios y Parámetros" permite al usuario configurar los costos de sus servicios. Una opción destacada es "Precios por Tipo de Aplicación", que al seleccionarse, abre una sub-pantalla (PricesScreen) dedicada donde se pueden establecer los precios por hectárea para aplicaciones líquidas, sólidas, mixtas y varias, cada uno editable mediante un diálogo numérico. Otra preferencia en esta sección es "Gestionar Formulaciones", que redirige al usuario a una pantalla separada para organizar y crear nuevas composiciones de productos.Finalmente, la sección de "Información" proporciona detalles sobre la propia aplicación. Aquí, la opción "Sobre esta App" despliega un diálogo informativo (AboutDialog) que explica el propósito general de la aplicación, destacando sus funcionalidades clave como la gestión de trabajos y clientes, planificación precisa, control de ejecución, administración financiera y opciones de personalización, todo ello presentado con tarjetas de características individuales para una mejor comprensión.La pantalla hace uso extensivo de diálogos contextuales para la entrada de datos (como precios y tasas de cambio) y para la selección de opciones (como la moneda), asegurando una interacción clara y guiada para el usuario. En resumen, SettingsScreen es una interfaz robusta y fácil de usar, diseñada para brindar al usuario un control total sobre las configuraciones cruciales de la aplicación.
* **FormulacionesScreen:** Esta pantalla, denominada FormulacionesScreen, es la interfaz clave para la gestión y ordenamiento de las formulaciones o mezclas dentro de la aplicación. Su objetivo principal es permitir a los usuarios establecer el orden preciso en que los componentes deben ser añadidos a una mezcla, lo cual es fundamental para procesos posteriores como la preparación de recetas, además de proporcionar herramientas completas para añadir, editar y eliminar formulaciones existentes.Al ingresar a la pantalla, se presenta una barra superior que muestra el título "Ordenar Formulaciones" y un icono de ayuda. Al pulsar este icono, se despliega un diálogo específico (HelpDialog) que guía al usuario, explicando cómo la pantalla facilita la organización de los ingredientes mediante la funcionalidad de arrastrar y soltar, cómo gestionar las formulaciones individualmente, y destacando la importancia de este ordenamiento para la pantalla de "Recetas", donde se utiliza automáticamente para instruir al operario.La parte superior de la interfaz incluye un encabezado visualmente atractivo (FormulacionesHeader) que refuerza el propósito de la pantalla, mostrando un icono de ciencia y un texto indicando "Orden de Mezcla" y la acción de "Arrastra para reordenar formulaciones".El corazón de la pantalla es una lista desplazable (LazyColumn) que presenta cada formulación como un elemento interactivo (FormulacionItem). Cada ítem muestra claramente el número de orden de mezcla actual de la formulación, su nombre y su tipo (líquido o sólido), representado con iconos distintivos. La característica central es la capacidad de arrastrar y soltar: al realizar una pulsación larga sobre una formulación, el usuario puede arrastrarla a una nueva posición en la lista. Durante el arrastre, el elemento se resalta visualmente, y se muestra un indicador delgado en la posición donde se soltará el elemento, facilitando la reordenación. Este proceso de arrastre y soltar es gestionado por un estado (DragDropState) que sigue el movimiento del elemento y actualiza las posiciones.Además de la reordenación, cada formulación en la lista cuenta con botones de acción para "Editar" y "Eliminar". Al pulsar el botón de edición, se abre un diálogo (AddEditFormulacionDialog) que permite modificar el nombre y el tipo de unidad de la formulación, precargando los datos existentes si se está editando. Al seleccionar la opción de eliminar, se activa un diálogo de confirmación (DeleteFormulacionDialog). Este diálogo es inteligente, ya que primero verifica si la formulación está en uso en alguna otra parte de la aplicación (por ejemplo, en una receta); si lo está, impide la eliminación y muestra un mensaje de advertencia, asegurando la integridad de los datos.La pantalla también se integra con una acción flotante (Floating Action Button), que al ser pulsado, permite al usuario añadir una nueva formulación a través del mismo diálogo de edición, pero con campos vacíos. Finalmente, la pantalla incluye una funcionalidad de auto-guardado que se activa automáticamente cuando el usuario abandona la pantalla, asegurando que cualquier cambio en el orden o en las formulaciones se persista sin necesidad de una acción explícita de guardado. En su conjunto, FormulacionesScreen es una herramienta dinámica y esencial para la organización y gestión de los componentes de mezcla.
* **DocumentViewerScreen:** Esta pantalla, denominada DocumentViewerScreen, está diseñada para proporcionar una vista detallada de un "movimiento" o transacción financiera específico dentro de la aplicación, y para permitir al usuario interactuar con los documentos adjuntos a dicho movimiento. Su objetivo principal es ofrecer transparencia y acceso directo a la información y a los archivos relacionados con una entrada particular en el registro de movimientos.Al cargar, la pantalla muestra una barra superior (TopAppBar) con el título "Detalle del Movimiento" y un botón de retroceso (ArrowBack) que permite al usuario navegar a la pantalla anterior. La interfaz gestiona su estado de carga, mostrando un indicador de progreso circular (CircularProgressIndicator) mientras la información del movimiento y sus documentos asociados son recuperados. Si el movimiento no se encuentra, se presenta un mensaje claro al usuario indicando que el "Movimiento no encontrado".Una vez que el movimiento ha sido cargado exitosamente, la pantalla organiza la información de manera estructurada en una lista desplazable (LazyColumn). Primero, se muestran los detalles principales del movimiento: la descripción, que se presenta con un estilo de título, y la fecha del movimiento, formateada para ser fácilmente legible. Una línea divisoria horizontal separa esta información inicial de los detalles financieros. El monto del movimiento se calcula restando el "debe" del "haber" y se muestra prominentemente, formateado según la configuración de moneda de la aplicación y coloreado en verde (o el color primario del tema) si es un ingreso positivo, y en rojo (color de error) si es un egreso o saldo negativo. Además, si el movimiento incluye "detalles de pago" o notas, estos se presentan debajo del monto con un estilo de texto inclinado.Un componente crucial de esta pantalla es la sección de documentos. Si el movimiento tiene documentos adjuntos, se muestra un título "Documentos Adjuntos", seguido de una lista de estos documentos. Cada documento se representa visualmente con un DocumentoItem, que consiste en una tarjeta interactiva. Dentro de cada tarjeta, se muestra un icono genérico de descripción y el nombre del archivo del documento. La funcionalidad clave de esta sección es que, al pulsar sobre cualquiera de estas tarjetas de documento, la aplicación lanza una Intent para abrir el archivo con una aplicación externa del sistema compatible (por ejemplo, un visor de PDF para un archivo PDF, o una galería para una imagen), garantizando que el usuario pueda visualizar el contenido del documento directamente desde la aplicación. En resumen, DocumentViewerScreen proporciona una experiencia completa para revisar los detalles financieros y los documentos asociados a cada transacción.
* **ChecklistsScreen:** Esta pantalla, denominada ChecklistsScreen, es la interfaz central para la gestión integral de listas de verificación y sus respectivos elementos dentro de la aplicación. Su objetivo primordial es proporcionar un entorno intuitivo y visualmente dinámico para que los usuarios puedan crear, organizar, editar y realizar un seguimiento del progreso de sus tareas mediante checklists.Al iniciar, la pantalla puede mostrar un indicador de carga si los datos están siendo recuperados. Si no hay listas de verificación creadas, presenta un estado vacío que anima al usuario a crear su primera checklist, facilitando esta acción a través de un botón dedicado. Si ya existen checklists, estas se muestran en una fila horizontal deslizable en la parte superior, cada una representada por una tarjeta (ChecklistCard) que indica su título y un resumen de su progreso (elementos completados de un total). El usuario puede seleccionar cualquiera de estas tarjetas para ver y gestionar sus elementos en el área principal de la pantalla. Una barra superior animada muestra el título general "Checklists" y, si hay una seleccionada, el título de la checklist activa, junto con un icono para acceder a un panel que muestra todas las checklists disponibles.La pantalla soporta la creación de nuevas listas de verificación mediante un diálogo que solicita un título. También permite renombrar o eliminar checklists existentes a través de diálogos de confirmación, asegurando que el usuario tenga control total sobre su organización. Un botón flotante se adapta dinámicamente: si no hay checklists, abre el diálogo para crear una nueva; si ya existen, abre un menú inferior (ChecklistBottomSheet) que lista todas las checklists, permitiendo la selección, creación, renombramiento o eliminación de cualquiera de ellas.Cuando una checklist está seleccionada, la pantalla revela una sección dedicada a la gestión de sus elementos. Aquí, un campo de texto permite añadir nuevos ítems a la lista. Los ítems existentes se muestran en una columna vertical, donde cada uno puede ser marcado como completado o incompleto, editado (abriendo un diálogo específico para modificar su texto) o eliminado. Una característica destacada es la capacidad de reordenar los ítems de la lista mediante gestos de arrastrar y soltar, proporcionando una gran flexibilidad en la organización de las tareas. Si la checklist seleccionada no tiene ítems, se muestra un estado vacío que instruye al usuario a añadir su primer elemento.Toda la interacción está enriquecida con animaciones fluidas, como la aparición y desaparición de elementos, la transición entre checklists seleccionadas y la reordenación de ítems, lo que contribuye a una experiencia de usuario altamente responsiva y atractiva. Además, la pantalla utiliza mensajes temporales (Snackbar) para proporcionar retroalimentación sobre las acciones realizadas, como la creación, renombrado o eliminación de checklists. En conjunto, ChecklistsScreen es una herramienta robusta y visualmente interactiva para la planificación y el seguimiento de tareas detalladas.
* **LocationDialogs.kt**: Este archivo, LocationDialogs.kt, agrupa dos componentes cruciales en forma de diálogos para la interacción con ubicaciones geográficas dentro de la aplicación, utilizando la integración de Google Maps. Su objetivo principal es ofrecer una manera estandarizada y reutilizable de seleccionar y visualizar coordenadas, adaptándose a distintas necesidades de la interfaz de usuario.El primer componente es LocationPickerDialog, diseñado para permitir al usuario seleccionar una ubicación específica en un mapa. Este diálogo se presenta como una ventana emergente que contiene un mapa de Google. Inicialmente, el mapa se centra en una ubicación predefinida (si se está editando una ubicación existente) o en una vista más amplia. Si la pantalla se invoca en modo de edición, un marcador ya se encuentra en la posición inicial; de lo contrario, el mapa está listo para que el usuario interactúe. El usuario puede tocar cualquier punto del mapa para colocar un marcador, que indica la ubicación seleccionada. Este marcador puede moverse si se toca otra parte del mapa. Un botón de "Confirmar ubicación" se activa una vez que se ha seleccionado un punto, permitiendo al usuario guardar las coordenadas de latitud y longitud elegidas, lo que se traduce en una acción de guardado en la aplicación. También se incluye un botón para "Cancelar", que simplemente cierra el diálogo sin guardar cambios.El segundo componente es LocationViewerDialog, cuya función es mostrar una ubicación previamente establecida en el mapa. Similar al anterior, se presenta como un diálogo que contiene un mapa de Google. En este caso, el mapa se centra directamente en las coordenadas de latitud y longitud proporcionadas, mostrando un marcador con una descripción o título sobre la ubicación del trabajo. A diferencia del selector, este diálogo está enfocado en la visualización, pero ofrece un botón "Editar" que permite al usuario iniciar una acción para modificar la ubicación, lo que normalmente llevaría a abrir el LocationPickerDialog. También incluye un botón "Cerrar" para simplemente descartar la vista.En resumen, este archivo centraliza la lógica de interacción con mapas en diálogos modales, facilitando tanto la entrada de nuevas ubicaciones de manera interactiva como la visualización clara de coordenadas ya existentes, con la opción de editarlas. Esto asegura una experiencia de usuario consistente y eficiente para todas las funcionalidades de la aplicación que requieran manejo de información geográfica.
* **MainActivity.kt**: Este archivo, MainActivity.kt, actúa como el punto de entrada principal y el orquestador de la interfaz de usuario de la aplicación. Su objetivo fundamental es inicializar la aplicación, gestionar la experiencia de la pantalla de bienvenida (splash screen) y construir la estructura base de la interfaz de usuario con su barra superior, barra de navegación inferior y el contenido de las pantallas.Al inicio, el MainActivity integra la funcionalidad de la splash screen del sistema, asegurando que esta se mantenga visible mientras la aplicación carga los datos iniciales necesarios, evitando así una pantalla en blanco. Una vez que los datos están listos, la aplicación procede a configurar su contenido Compose.El corazón de la interfaz se construye alrededor del composable MainAppScaffold, que emplea un Scaffold de Material 3 para estructurar la aplicación. En la parte superior, una CenterAlignedTopAppBar muestra el título "Al Lote", incluye un icono de "Home" que aparece condicionalmente (solo si no se está en el Dashboard) para regresar al inicio, y un icono de "Configuración" que lleva a la pantalla de ajustes de la aplicación.En la parte inferior, una NavigationBar personalizada sirve como el principal método de navegación entre las secciones clave de la aplicación. Esta barra contiene NavigationBarItems para "Trabajos" (que incluye una insignia o "badge" mostrando el número de trabajos pendientes), "Clientes", "Admin" y "Productos". Una característica distintiva de esta navegación es la inclusión de un botón flotante de acción (FAB) integrado directamente en el centro de la barra inferior, que permite al usuario realizar acciones principales como agregar un nuevo elemento, y cuya función exacta puede ser definida por la pantalla actual a través de un callback. La visibilidad de esta barra inferior es dinámica, ocultándose cuando el usuario está en la pantalla del Dashboard.El contenido principal de la aplicación se maneja a través de un AppNavHost, que es el encargado de gestionar la navegación entre las distintas pantallas (como trabajos, clientes, admin, productos, dashboard, etc.) basándose en la ruta actual. Este AppNavHost también proporciona los callbacks necesarios para que el MainAppScaffold pueda controlar la acción del botón flotante y mostrar un diálogo de "Agregar Trabajo".El MainActivity también se encarga de la gestión de diálogos modales. Específicamente, controla la aparición de un JobDialogCompose para agregar nuevos trabajos. Antes de mostrarlo, verifica si existen clientes registrados; si no los hay, un AlertDialog informativo guía al usuario a la pantalla de clientes antes de permitirle crear un trabajo. Una vez que un trabajo es guardado a través del diálogo, se muestra un mensaje breve (Toast) al usuario confirmando la acción. En resumen, MainActivity.kt es el cimiento sobre el cual se construye toda la experiencia de usuario, proporcionando una estructura de navegación coherente, una gestión de estado centralizada y una interacción fluida con los componentes principales de la aplicación.

**OBSERVACIONES**

* Recordar que no sé mucho de programacion por lo que no es buena idea utilizar terminos o conceptos demasiado complejos de entender.
* Tampoco soy muy conocedor sobre la manera en que trabajan las inteligencias artificiales por lo que tienes libertad para decidir de que forma trabajar para que sea lo mas prolijo y eficiente posible.
* No conozco nada sobre las buenas practicas a la hora de desarrollar una app por lo que tambien dejo esto en manos de la IA.
* Ya que no conozco mucho sobre la programacion y sus metodos no será muy necesario que me des explicaciones detalladas de todo lo que se realizará ya que seguramente no entenderé mucho de esto y ademas asi se reducirá el uso innecesario de tokens.
* Todo lo que se realice se me debe explicar de una forma facil y sencilla de entender.
* Todo lo que se desarrolle debe ser pensando en que en algun momento la app se publicará en el PlayStore por lo que se debe cumplir con los requisitos necesarios para la correcta publicacion de la app.
* A medida que se desarrollen nuevas implementaciones podrás notar ciertos apectos que puedan mejorarse o pulirse. Si sucede eso puedes informarmelo y evaluaremos si realizar la mejora o correccion correspondiente.
* Posiblemente en el futuro se realizará una refactorizacion para que la app tambien pueda ser utilizada en dispositivos con sistema operativo iOs.