

# Notes

Desarrollo Avanzado de Software

*Proyecto II*

Paula de Jaime

# Índice

<b>Descripción de la aplicación</b>	<b>3</b>
<b>Clases utilizadas</b>	<b>5</b>
<b>Requisitos</b>	<b>9</b>
Requisitos Mínimos	9
Requisitos Opcionales	14
<b>Manual de Usuario</b>	<b>17</b>

## Descripción de la aplicación

[https://drive.google.com/file/d/1hvqvMv8QCK0\\_vAudhV\\_izyNCI7Qd5XKE/view?usp=sharing](https://drive.google.com/file/d/1hvqvMv8QCK0_vAudhV_izyNCI7Qd5XKE/view?usp=sharing)

Este informe es el entregado en la primera parte del proyecto con información añadida.

Para que esta aplicación funcione correctamente, será necesario tener *Google Play* instalado. También se recomienda tener *Google Drive*, *Google Calendar* y *Google Maps* para poder visualizar los resultados de las funcionalidades ejecutadas.

La aplicación *Notes* permite a los usuarios escribir, guardar y editar sus propias notas, evitando así que se les olvide cualquier información que deseen recordar.

Las funcionalidades principales serán:

- **Crear notas:** Los usuarios tendrán la posibilidad de hacer uso de varios estilos de texto (negrita, itálica y la posibilidad de incluir enlaces).
- **Editar notas:** Se podrán editar las notas anteriormente creadas.
- **Borrar notas:** Las notas creadas se podrán borrar.
- **Crear etiquetas:** Esta funcionalidad la podrá usar el usuario para clasificar las notas según le convenga. Un usuario no podrá tener dos etiquetas iguales, es decir, con el mismo nombre. Las etiquetas se podrán asignar mientras la nota se esté creando o editando. Una nota solo podrá tener asignada una etiqueta.
- **Borrar etiquetas:** Las etiquetas se podrán borrar sin que éstas afecten a las notas a las que están asignadas, es decir, estas notas se quedarán sin etiqueta.
- **Cambiar la configuración:** Los usuarios tendrán distintas opciones a configurar para mejorar su experiencia.
  - *Recibir notificaciones:* el usuario decidirá si desea recibir una notificación por cada nueva nota que cree. Esta funcionalidad es interesante, ya que las notificaciones servirían de recordatorio al permanecer constantes en la barra superior del móvil.

- *Mostrar notas recientes primero:* si esta opción se habilita, las notas más recientes aparecerán arriba en la pantalla principal.
- *Cambiar lenguaje:* el usuario podrá cambiar en todo momento el lenguaje de la aplicación mediante esta opción. Las opciones serán: sistema (el lenguaje por defecto del móvil), español e inglés.
- *Guardar posición de notas:* cuando el usuario crea una nota, puede elegir que su última posición en el mapa quede asignada a ésta. Estas posiciones se podrán visualizar en un mapa.
- **Ver mapa:** Esta funcionalidad permitirá al usuario visualizar donde creó sus notas. Se mostrarán las notas en el mapa mediante el uso de etiquetas.
- **Añadir nota al calendario:** Las notas se podrán añadir al calendario, en un día y hora específicos. Por defecto, estos eventos durarán dos horas.
- **Subir notas a Google Drive:** Las notas se podrán subir a Google Drive tras iniciar sesión en una cuenta de Google. Estas notas se guardarán automáticamente en una carpeta específicamente creada para esta aplicación.
- **Crear notas cortas:** Se crea una nueva modalidad de notas, las notas cortas o rápidas ("self notes"). Se suelen usar aplicaciones de mensajería instantánea para apuntarse cosas que se necesitan recordar. Esta funcionalidad planea dejar atrás este sistema, dejando que los usuarios "chateen" con ellos mismos. Se podrán enviar tanto texto como imágenes.

Para realizar cualquiera de las funcionalidades anteriores, el usuario deberá de haber iniciado sesión. Todas las notas y etiquetas creadas por un usuario no podrán ser vistas por el resto de miembros registrados en la aplicación.

## Clases utilizadas

Se han utilizado diferentes actividades y clases:

- **MyDB:** Clase que se encarga de crear la base de datos junto con sus tablas. Todas las consultas/actualizaciones de la base de datos ocurren dentro de esta clase mediante el uso de funciones.
- **LoaderWelcomeActivity:** Esta actividad es la entrada de la aplicación, es decir, es la primera que se carga. Mediante el uso de *Threads* se ha creado una especie de pantalla *Splash* que desaparecerá tras 1500 milisegundos (1.5 segundos aproximadamente).

El usuario será redirigido a la pantalla de inicio de sesión si no se ha encontrado en la base de datos a un usuario activo, es decir, que haya iniciado sesión previamente y que no la haya cerrado.

Si se encuentra a un usuario activo, dicho usuario se guardará en la clase **Data**. Dicha clase existe para que no se tenga que consultar la base de datos repetidamente si se quiere saber cuál es el usuario activo.

- **LoginActivity:** Como bien su nombre indica, esta actividad se encargará de la lógica del inicio de sesión. Como en el caso anterior, si un usuario inicia sesión satisfactoriamente, su nombre de usuario se guardará en la clase **Data**.
- **SignUpActivity:** Esta actividad será la encargada de registrar a los usuarios en el sistema. Se mostrará un error si el nombre de usuario ya existe.
- **LanguageActivity:** Esta actividad se encarga de cargar el idioma definido por el usuario, haciendo uso de las preferencias para saber cuál ha sido el elegido. Se ejecuta dentro de *onCreate*, y gracias al uso de la herencia, se ejecuta antes de cada actividad logrando así que el idioma se cambie correctamente. Un cambio de orientación no afecta a esta actividad, ya que el idioma se carga en *onCreate*.
- **Common:** Esta actividad contendrá funcionalidades comunes como *showToast*. Heredará de **LanguageActivity**, y se ocupará de crear un fragmento que contendrá cualquier tarea asíncrona. Esto ocurre para que dichas tareas no dejen de ejecutarse al girar el dispositivo.

- **MainToolbar:** Clase que implementa la lógica del menú superior. Como todas las actividades heredan de esta clase, se ha hecho que esta clase a su vez herede de **Common**. De esta manera conseguimos que todas las pantallas sean afectadas por el cambio del lenguaje.

Las funcionalidades compartidas (borrar y editar nota, compartir por correo, subir nota a drive,...) por varias de las actividades que heredan esta clase se han implementado aquí para reducir la duplicidad de código.

- **MainActivity:** Esta actividad controlará la pantalla que el usuario visualizará una vez que haya iniciado sesión. Al hacer uso de *fragments* decidirá si debe abrir otra actividad o simplemente cargar los datos correspondientes cuando se seleccione una nota.

Será la responsable de la creación de notas, iniciando de esta manera la actividad correspondiente, aparte de permitir gestionar (borrar y crear) las etiquetas existentes.

La interfaz gráfica cargada por esta actividad será la del listado de notas disponibles del usuario.

Esta actividad está relacionada con las clases **NotesFragment**, **ElViewHolder** y **ElAdaptadorRecycler**. Las dos últimas dos clases serán las encargadas de crear el *RecyclerView+CardView*. La primera obtendrá los datos correspondientes de las notas para que el *RecyclerView+CardView* los cargue, aparte de contener los métodos necesarios para actualizar estos datos posteriormente.

- **SingleNoteActivity:** Actividad encargada de la visualización del contenido de una nota por pantalla. Está relacionada con **SingleNoteFragment** (uno de los dos fragmentos contenidos por **MainActivity** en landscape), clase encargada de cargar el contenido en el elemento gráfico correspondiente.
- **NoteEditorActivity:** Actividad que contiene la lógica del editor de textos utilizado para la creación y edición de las notas. Mediante el uso de un atributo se distingue si la nota es nueva o se está editando, evitando así la necesidad de dos actividades diferentes.

Esta clase también se encarga de controlar los estilos que el usuario selecciona (negrita, cursiva e inserción de enlaces) gracias a la clase

***SpanStyleHelper***, aparte de permitir la creación de nuevas etiquetas y la asignación de una de estas a la nota.

La clase ***SpanStyleHelper*** decide si el texto seleccionado debe ser negrita (o *itálica*) o no, es decir, si un texto ya es negrita y se vuelve a seleccionar negrita, el texto volverá a tener un estilo normal.

- ***PreferencesActivity***: Actividad contenedora del ***PreferencesFragment***. Esta última clase, permite cargar el fichero de configuración necesario para que se le muestre por pantalla al usuario. También se encarga de capturar los eventos surgidos cuando el usuario ha cambiado la configuración de algún elemento.
- ***AsyncTaskFragment***: Esta clase es un fragmento que se ocupará de ejecutar cualquier tarea asíncrona, ya sea, iniciar sesión, obtener las últimas notas cortas,... La tarea asíncrona a realizar se determinará mediante el atributo *action*. Como se ha comentado anteriormente, dichas tareas no detendrán su ejecución al girar la pantalla, ni al cerrar la aplicación.
- ***NotesToSelf***: Esta actividad se encargará de gestionar la carga y creación de las notas cortas. También gestionará los permisos correspondientes de la cámara.

Esta actividad está relacionada con las clases ***ElAdaptadorRecyclerSelfNotes***, y ***ElViewHolderSelfNote***. Estas dos clases serán las encargadas de crear el *RecyclerView+CardView*. Al contrario que el *RecyclerView* anterior, éste tendrá dos plantillas, una para las imágenes y otra para el texto.

- ***ServicioFirebase***: Servicio que contendrá la lógica para la llegada de mensajes del sistema Firebase.
- ***DriveServiceHelper***: Actividad obtenida de los ejemplos de Android para facilitar la comunicación con la api de Google Drive. Se ha modificado el código añadiendo nuevas funciones.
- ***MapsActivity***: Actividad encargada de cargar el mapa de Google Maps en pantalla, además de añadir las etiquetas correspondientes a las notas creadas.

- **UploadToDriveService:** Servicio encargado de subir una nota a Google Drive. Hay que tener en cuenta que si la aplicación se cierra (se quita de aplicaciones recientes), la subida de la nota se cancelará.
- **WidgetTema18:** Clase que se encarga de gestionar el widget de la aplicación. Gestiona su creación y actualización.

En algunas de las clases anteriores se han mencionado ciertas acciones que requieren el uso de *dialogs*. Los *dialogs* existentes se encuentran dentro de un paquete específico que contiene:

- **AddRemoveTag:** Se utiliza para añadir o borrar las etiquetas existentes. Esta funcionalidad solo está disponible desde el menú superior en la **MainActivity**.
- **ConfirmExit:** Mensaje de confirmación de que un usuario quiere salir (volver para atrás) de la pantalla actual. Este mensaje se utiliza en **NoteEditorActivity** para que el usuario no pierda los cambios realizados en la nota si se ha equivocado de botón.
- **DeleteNoteDialog:** Mensaje de confirmación antes de borrar una nota.
- **DeleteTextStyles:** Mensaje de confirmación antes de borrar los estilos de un texto. Los estilos hacen referencia a texto en negrita, itálica y enlaces.
- **InsertLinkEditor:** Se utiliza para que el usuario inserte el link con el texto a mostrar por pantalla. Este *dialog* se utiliza en **NoteEditorActivity**.
- **NewTag:** Pantalla que permite al usuario introducir el nombre de la etiqueta que desea crear.
- **SelectTagEditor:** Permite al usuario (de)seleccionar la etiqueta asignada a una nota dentro de **NoteEditorActivity**.
- **DateDialog:** Permite al usuario seleccionar una fecha. Se usa dentro de **MainToolbar**.
- **TimeDialog:** Permite al usuario seleccionar una hora. Se usa dentro de **MainToolbar**.



# Requisitos

## Requisitos Mínimos

- *Uso de una base de datos remota para el registro y la identificación de usuarios.*
  - Los usuarios cuando se registran o inician sesión lo hacen contra una base de datos remota.
  - Las notas cortas se guardan en una base de datos remota y local para dar soporte offline.
- *Uso de mensajería FCM.*

La mensajería FCM se usa en las notas cortas (nueva nota -> nota corta). Si un usuario ha iniciado sesión en dos dispositivos diferentes, se pueden dar dos casos:

- El primer caso es que se envíe la nota y el segundo dispositivo también esté en la pantalla de notas cortas. Si este es el caso, la nota se añadirá automáticamente (Fig. 1 y Fig. 2).

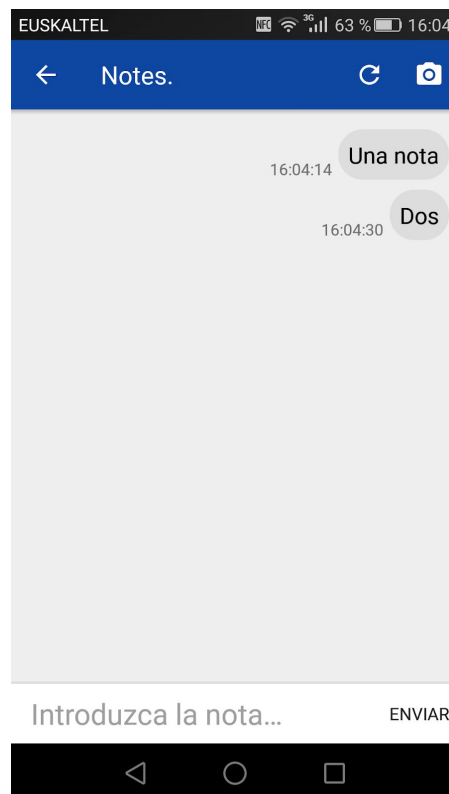


Fig. 1 - Dispositivo que recibe la nota

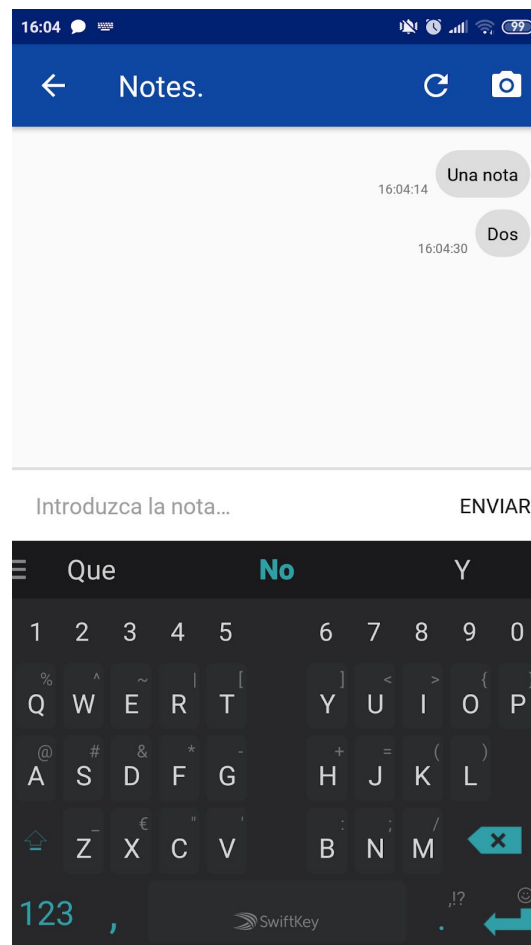


Fig. 2 - Dispositivo que envía la nota

- En el segundo caso, se envía la nota y el segundo dispositivo tiene la aplicación en segundo plano o cerrada. Como bien se muestra en las figuras 3 y 4, aparece una notificación indicando nuevas notas.

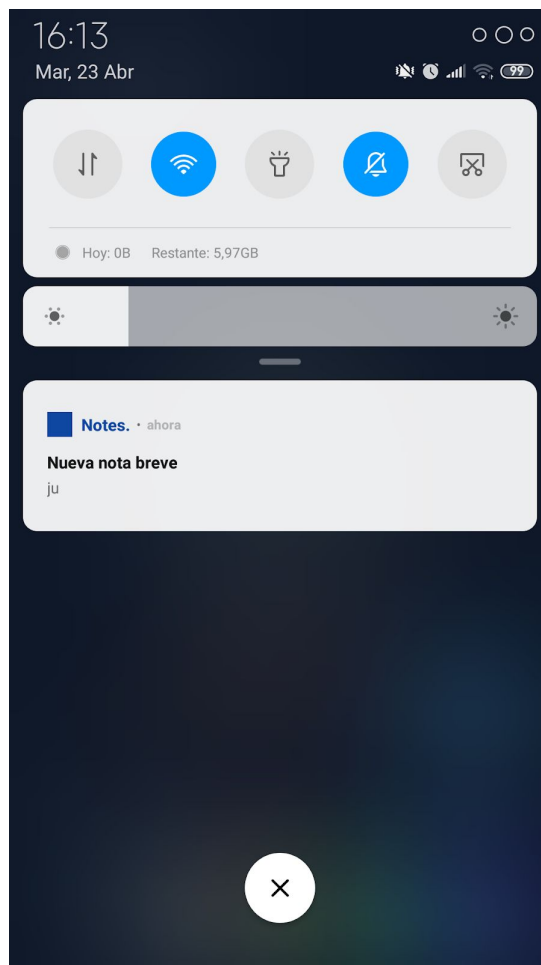


Fig. 3 - El dispositivo recibe una notificación

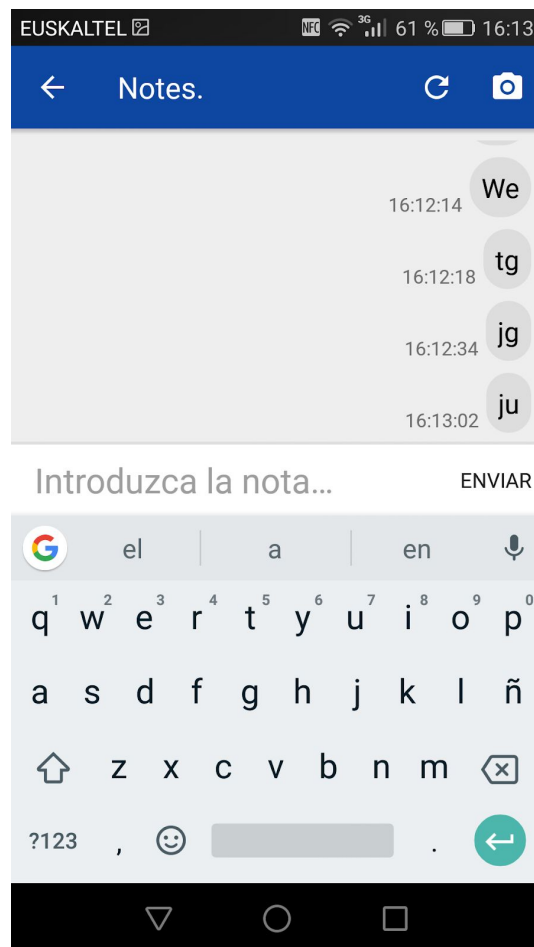


Fig. 4 - El dispositivo envía la nota

Se han dado ocasiones en las que los mensajes se retrasan haciendo creer que no llegan o no van a llegar.

- *Captar imágenes desde la cámara, guardarlas en el servidor y mostrarlas en la aplicación.*

La actividad de notas cortas permite enviar fotos desde la cámara. Estas fotos se guardan en un servidor remoto, y se descargan cada vez que se accede a la actividad de notas cortas, se hace click en actualizar (menú superior), o se recibe un mensaje estando en la actividad de notas cortas (Fig. 5 y Fig. 6).

Por cada foto descargada se guardarán dos archivos, la foto con la calidad original, y una para mostrar en las *ImageViews* del *RecyclerView*.

Cuando se haga click en una imagen, se le pedirá al usuario que elija la aplicación con la que la quiere abrir. La imagen abierta tendrá la calidad

original de la foto enviada, es decir, no será la miniatura utilizada en los *ImageViews*.

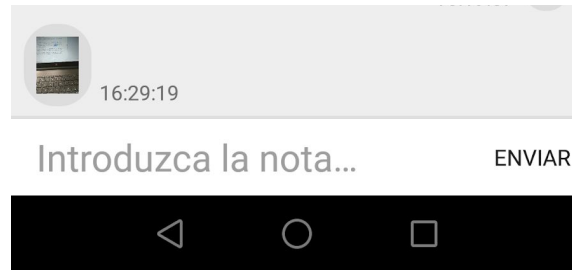


Fig. 5 - Dispositivo que recibe la foto

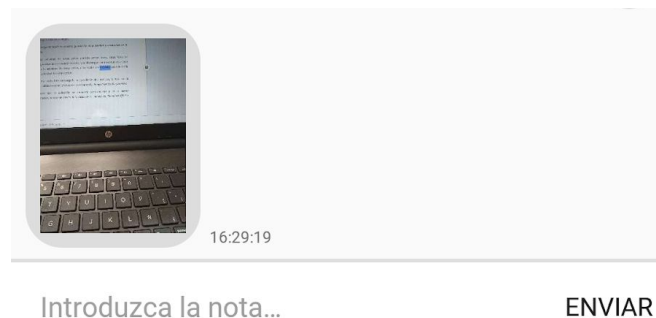


Fig. 6 - Dispositivo que manda la foto

- *Controlar que la aplicación se comporte correctamente y no se pierda información, aunque en medio de la ejecución recibamos una llamada telefónica, giremos el móvil, etc. No se puede impedir que la aplicación se gire. Controlar la pila de actividades para evitar incoherencias.*
  - Se ha probado que la información es consistente girando la pantalla.
  - Se ha controlado la pila de actividades para evitar incoherencias. Por ejemplo:
    - Cuando el usuario cierra sesión se limpia la pila de actividades.
    - En las notificaciones, en los *pendingIntents* se ha creado un stack personalizado mediante el *TaskStackBuilder*, para que si un usuario sigue la acción de una notificación, la pila de actividades sea coherente.
    - Se ha tenido sumo cuidado en la creación de nuevas actividades utilizando en ocasiones *finish* para evitar incoherencias.

- *Documentar la aplicación mediante comentarios en el propio código.*
  - Cada función está documentada.
  - Se han ido añadiendo comentarios para clarificar el código.

## Requisitos Opcionales

- *Uso de algún content provider para añadir, modificar o eliminar datos.*
  - Se utiliza el content provider del calendario para agregar una nota a éste. La nota se añadirá al calendario por defecto. (Fig. 7 y Fig. 8)



Fig. 7 - Elegir fecha para añadir la tarea al calendario

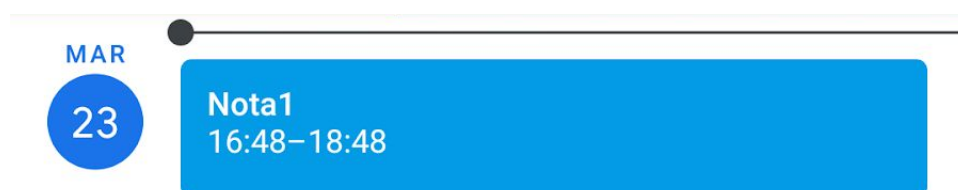


Fig. 8 - Nota añadida al calendario

- *Uso de algún servicio de Google (Geolocalización, Drive, Analytics, Calendar, etc.).*
  - Se ha utilizado el servicio *Firebase* en los mensajes FCM.

- Las notas se pueden subir a *Google Drive* (Fig. 9 y Fig. 10).

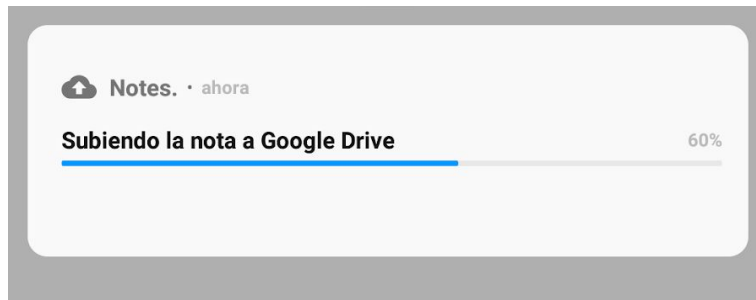


Fig. 9 - Notificación de que la nota se está subiendo.

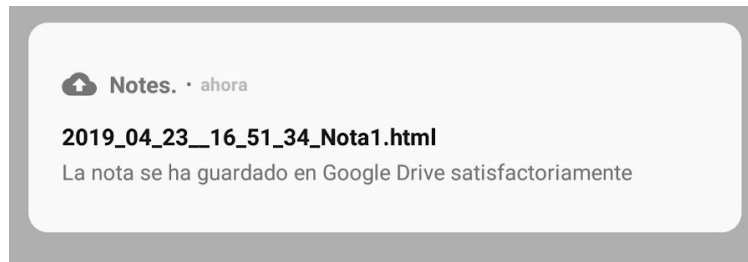


Fig. 10 - Notificación de que la nota se ha subido.

- *Uso de una actividad de Google Maps con etiquetas.*

Las notas se visualizarán en el mapa si se activa la funcionalidad correspondiente en *configuración*. (Fig. 11)



Fig. 11 - Etiqueta donde se ha creado cada nota

- *Desarrollar un widget que tenga, al menos, un elemento que se actualice automáticamente de manera periódica.*

El widget creado, una vez que el usuario ha iniciado sesión, indica el número de notas que ha creado. Se actualiza cada media hora, y también cuenta con un botón para actualizarlo manualmente.

- *Uso de algún servicio, tarea asíncrona o trabajo que no sea estrictamente necesario.*

Las siguientes funcionalidades se realizan mediante tareas asíncronas:

- Iniciar sesión
- Registrarse
- Cerrar sesión
- Actualizar notas cortas
- Enviar nota corta
- Enviar foto como nota corta
- Descargar fotos

En cambio, subir notas a Google Drive se consigue mediante un servicio.



## Manual de Usuario

Cuando el usuario abre la aplicación por primera vez se le mostrará la pantalla de inicio de sesión (Fig. 13). Si el usuario todavía no está en el sistema y quiere registrarse pulsará en “Regístrate”.

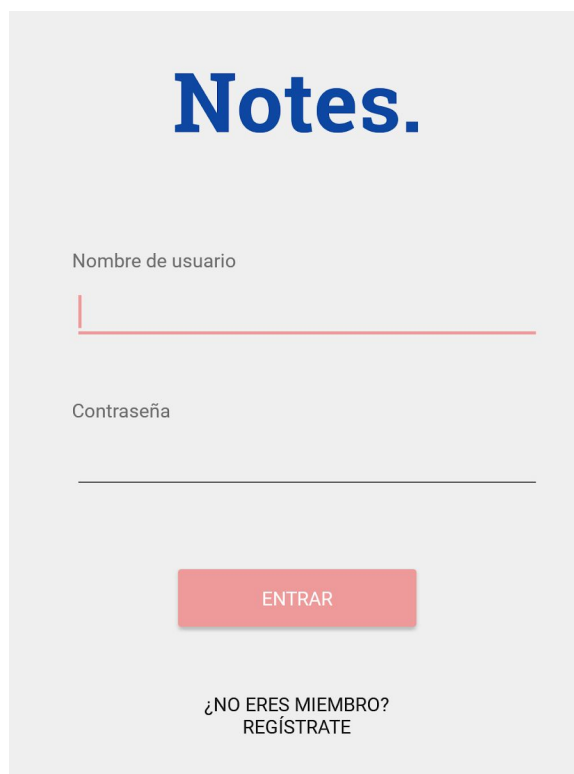
La imagen muestra la interfaz de inicio de sesión de una aplicación llamada "Notes.". El título "Notes." está en la parte superior en una fuente azul grande. Debajo hay dos campos de entrada: "Nombre de usuario" con un borde rojo y "Contraseña" con un borde gris. Entre los campos hay un espacio. Debajo de los campos hay un botón rojo con el texto "ENTRAR" en blanco. En la parte inferior, centrado, hay el texto "¿NO ERES MIEMBRO?" y "REGÍSTRATE" en una fuente más pequeña.

Fig. 13 - Inicio de sesión

Tras haber iniciado sesión (Nombre de usuario: “admin”, Contraseña: “1111”), el usuario podrá ver las notas que ha creado anteriormente (Fig. 14). Cuando gire la pantalla, la visualización de estas cambiará (Fig. 15).

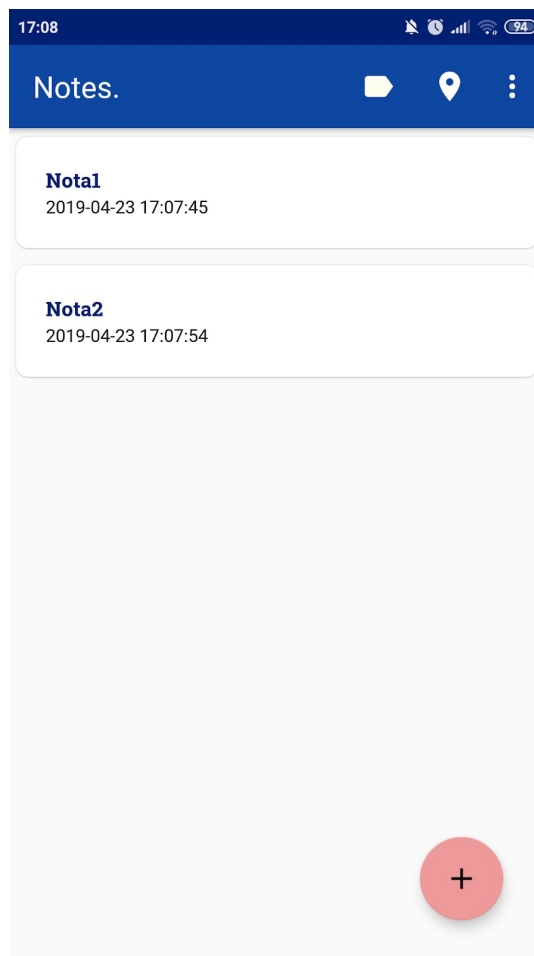


Fig. 14 - Notas del usuario (I)

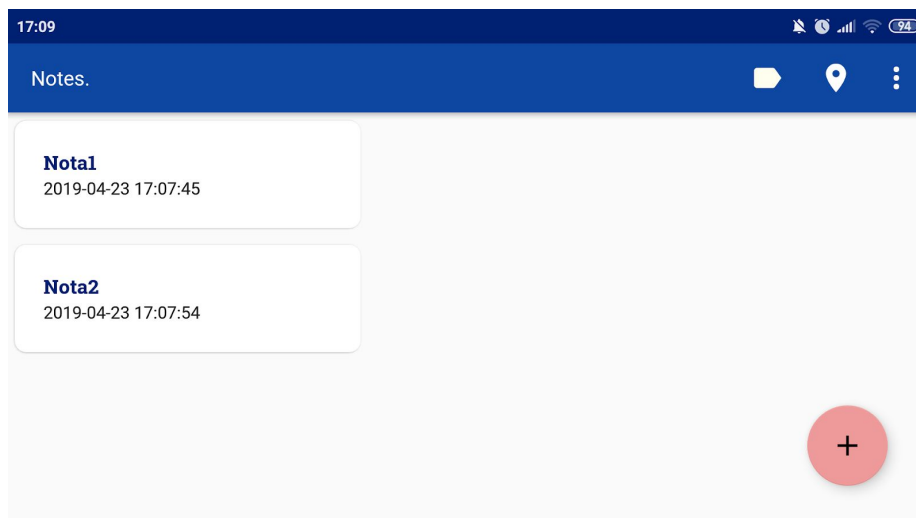


Fig. 15 - Notas del usuario (II)

Si se selecciona el botón de la etiqueta en la barra superior se podrán crear y eliminar las etiquetas del usuario (Fig. 16).

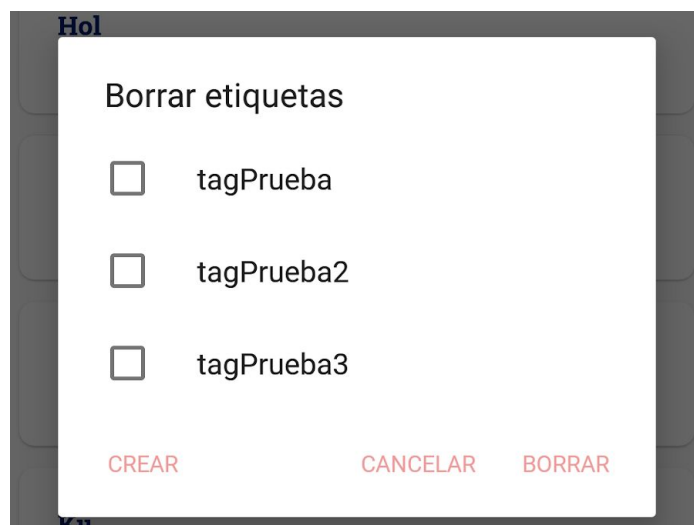


Fig. 16 - Crear o borrar etiquetas

Si en la barra superior se selecciona el icono del mapa, se mostrará un mapa con las etiquetas de donde se crearon las notas (si hay) (Fig. 11).

Mediante los tres puntos de la barra superior se puede cerrar sesión y acceder a la configuración de la aplicación (Fig. 17).

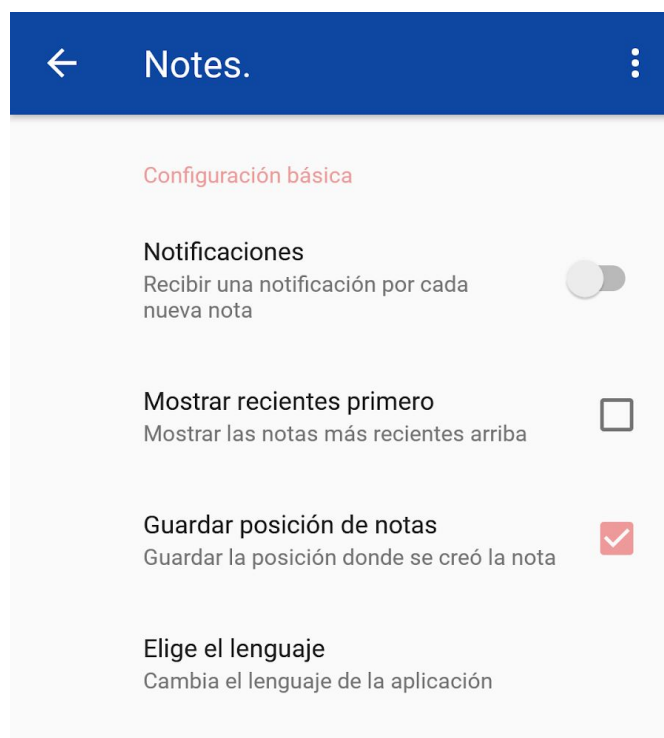


Fig. 17 - Configuración de la aplicación

Por otro lado, si se pulsa en el botón rosa de la esquina inferior derecha (Fig. 14) se abrirá una ventana donde se podrá elegir el tipo de nota a crear (Fig. 18)

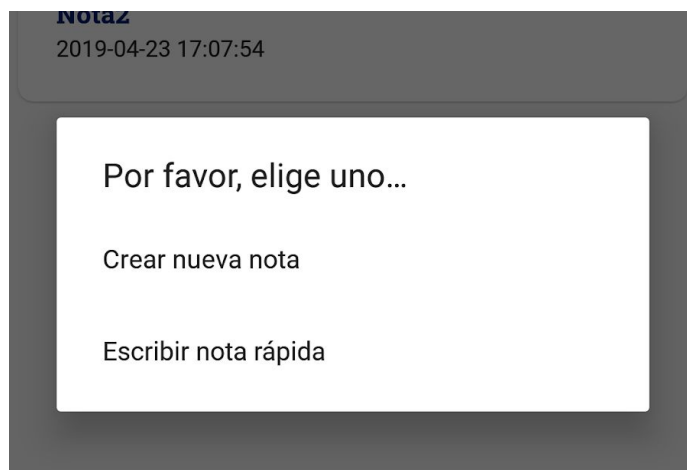


Fig. 18 - Elegir el tipo de nota a crear

Si se elige "Crear nueva nota", aparecerá el editor de las notas (Fig. 19).

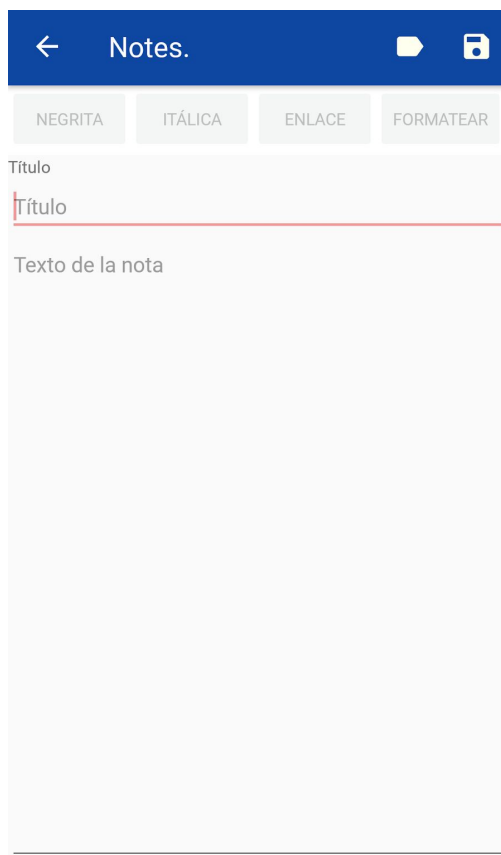


Fig. 19 - Editor de notas

Cuando se haga clic en “Añadir etiqueta” se abrirá una ventana que permitirá seleccionar la etiqueta a asignarle a la nota (Fig. 20).

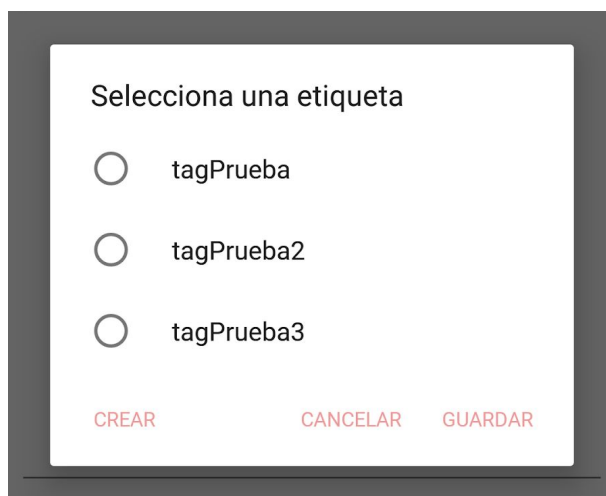


Fig. 20 - Asignar una etiqueta a la nota

Si se desea desasignar una etiqueta tan solo habrá que deseleccionar la etiqueta pertinente y guardar.

Los botones superiores de edición (Fig. 19) solo estarán disponibles cuando se esté editando el cuerpo de la nota.

Cuando el usuario seleccione una de las notas en la Fig. 14 o Fig. 15, el contenido de esta se visualizará por pantalla y la barra de arriba contendrá diferentes acciones (Fig. 21 y 22).

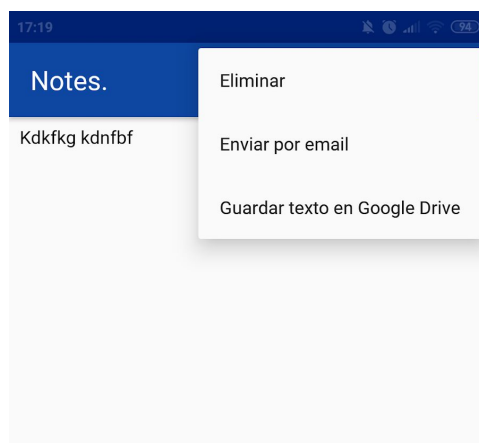


Fig. 21 - Visualización de una nota

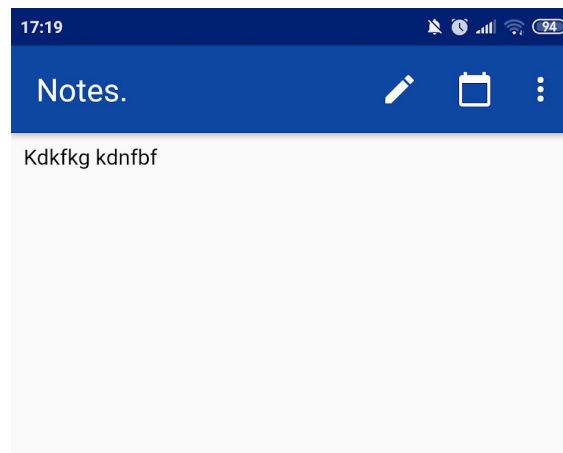


Fig. 22 - Visualización de una nota II

El botón de eliminar visto en la Fig. 21 permitirá al usuario borrar la nota, mientras que el botón del sobre servirá para enviar por correo dicha nota.

Cuando se pulse en el lápiz, se abrirá el editor de notas (Fig. 19) para poder modificar la nota.

El botón del calendario permitirá añadir la nota al calendario por defecto.

Si por el contrario en la figura 18 seleccionamos escribir una nota rápida, se abrirá una pantalla parecida a la de un chat (Fig. 23).

Las imágenes aparecerán a la izquierda, y los mensajes de texto a la derecha.

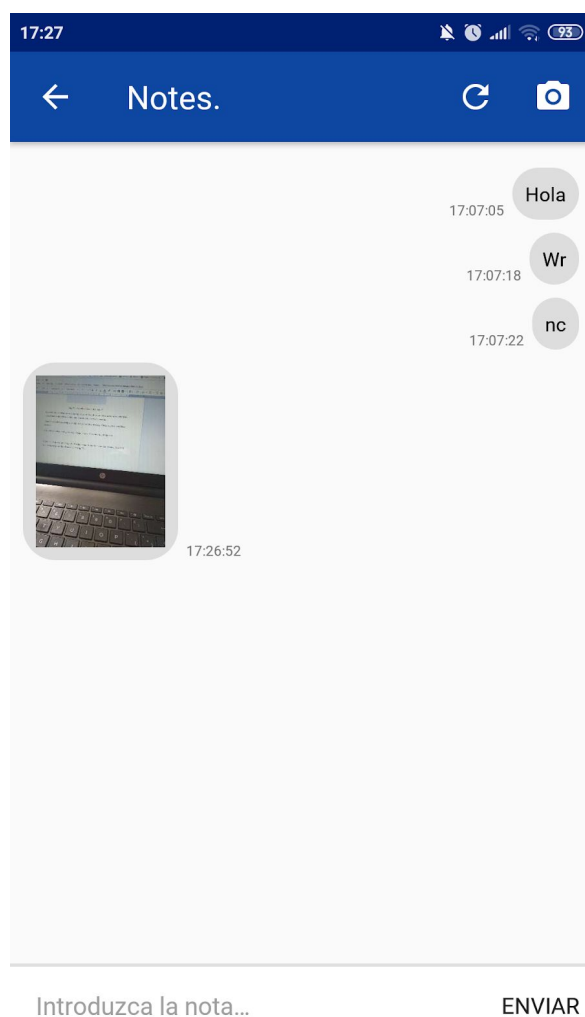


Fig. 23 - Pantalla de notas rápidas/cortas

Si el usuario se decide a añadir el widget de la aplicación (Fig. 24), podrá ver el número de notas que ha escrito siempre que haya iniciado sesión. Se actualizará cada media hora.

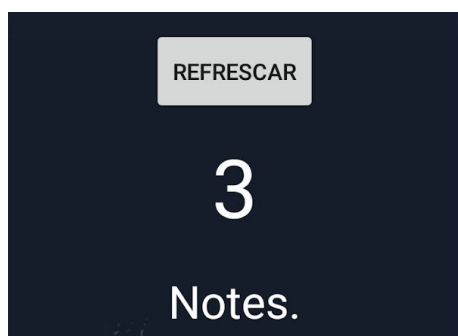


Fig. 24 - Widget