

# **TRABAJO DE LA ASIGNATURA AMPLIACIÓN DE LA APLICACIÓN DE NOTAS**

Ingeniería del Software 2019 - 2020

Pedro Tamargo Allué - 758267  
Juan José Tambo Tambo - 755742

# Tabla de contenidos

<b>Tabla de contenidos</b>	<b>1</b>
<b>Tabla de requisitos completa</b>	<b>2</b>
Requisitos Funcionales	2
Requisitos No Funcionales	3
Diagrama CU y descripción textual de los mismos	3
Flujo de eventos de casos de uso	4
<b>Análisis</b>	<b>6</b>
Diagramas de clases de análisis	6
Diagramas de secuencia para los Casos de Uso	6
Descripción de la interfaz de usuario	11
Prototipos de las nuevas pantallas y mapa de navegación	11
<b>Diseño del sistema</b>	<b>12</b>
Descomposición del sistema en diagramas de paquetes	12
Diagrama de despliegue	12
Distribución en componentes	13
<b>Diseño de objetos</b>	<b>13</b>
Decisiones tomadas	13
Patrón Bridge para envío de notas	13
Patrón Adapter para el acceso a la Base de Datos	14
Diagramas con mayor nivel de detalle	15
Modelos estáticos	15
Modelos dinámicos	15
Modelo E-R de la BBDD	20
<b>Pruebas</b>	<b>21</b>
Pruebas unitarias y de caja negra	21
Clases de equivalencia: updateNote(long rowId, String title, String body)	21
Resultado de ejecución de los tests	21
Prueba de volumen	21
Prueba de sobrecarga	22
Pruebas ejecutadas en el emulador	22
Pruebas ejecutadas con monkey	22
<b>Resultados y conclusiones</b>	<b>23</b>

## Tabla de requisitos completa

### Requisitos Funcionales

Identificador	Requisito
RF1	El usuario puede crear una nota
RF2	El usuario puede modificar una nota existente
RF3	El usuario puede eliminar una nota existente
RF4	El usuario puede consultar el listado de notas creadas
RF5	Las notas están compuestas de un título y de un cuerpo. El usuario no podrá crear una nota sin título.
RF6	El usuario podrá enviar una nota por SMS
RF7	El usuario podrá enviar una nota por correo electrónico
RF8	El usuario puede crear una categoría
RF9	El usuario puede modificar una categoría
RF10	El usuario puede eliminar una categoría
RF11	El usuario puede asignar una categoría a una nota previamente creada
RF12	El usuario puede eliminar la categoría previamente asignada a una nota.
RF13	El usuario puede ordenar el listado de notas por categoría
RF14	El usuario puede ordenar el listado de notas por título
RF15	El usuario puede mostrar solamente las notas de una determinada categoría
RF16	El usuario puede enviar la nota mediante un mail o SMS

## Requisitos No Funcionales

Identificador	Requisito
RNF1	No se espera que haya más de 1000 notas almacenadas en la aplicación.
RNF2	No se espera que haya más de 1000 categorías almacenadas en la aplicación.

## Diagrama CU y descripción textual de los mismos

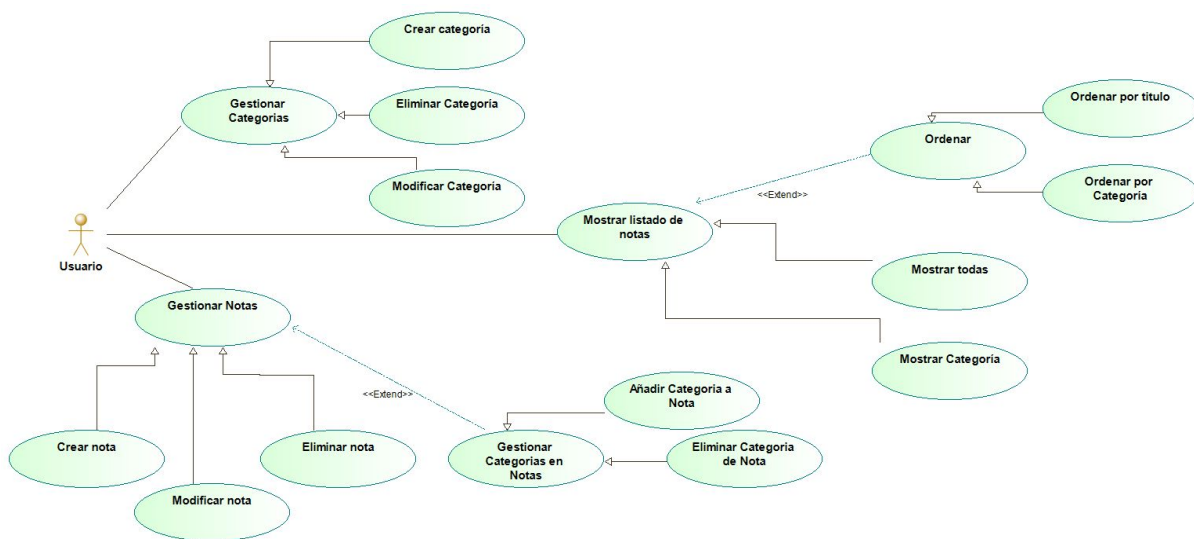


Diagrama de casos de uso para el sistema especificado anteriormente.

## Flujo de eventos de casos de uso

### Crear Nota (Flujo ppal)

- 1) Caso de uso se inicia cuando el usuario solicita crear nota.
- 2) El usuario proporciona un título a la nota y texto para el cuerpo de la misma.
- 3) El sistema guarda la nota.

### Modificar nota (Flujo ppal)

- 1) Caso de uso se inicia cuando el usuario selecciona una nota existente.
- 2) El usuario modifica el título, cuerpo o ambos de la nota seleccionada.
- 3) El sistema guarda la nota.

#### Eliminar nota (Flujo ppal)

- 1) El caso de uso se inicia cuando se selecciona una nota existente y se solicita su borrado.
- 2) El sistema borra la nota.

#### Enviar nota (Flujo ppal)

- 1) El caso de uso se inicia cuando se selecciona una nota existente y se solicita su envío.
- 2) El sistema inicia la ventana de envío, ya sea utilizando mail (si el tamaño de la misma supera los 100 caracteres) o SMS (si su tamaño es menor a 100 caracteres).
- 3) El usuario selecciona la opción "enviar".

#### Listar notas (Flujo ppal)

- 1) El caso de uso se inicia al arrancar la aplicación.
- 2) El sistema recupera las notas existentes y muestra sus títulos.

#### Listar notas (Flujo alternativo)

- 1) El caso de uso se inicia cuando se acaba de crear una nota.
- 2) El sistema recupera las notas existentes y muestra sus títulos.

#### Listar notas (Flujo alternativo)

- 1) El caso de uso se inicia cuando se acaba de modificar una nota.
- 2) El sistema recupera las notas existentes y muestra sus títulos.

#### Listar notas (Flujo alternativo)

- 1) El caso de uso se inicia cuando se acaba de eliminar una nota.
- 2) El sistema recupera las notas existentes y muestra sus títulos.

#### Listar notas (Flujo alternativo)

- 1) El caso de uso se inicia cuando se acaba de enviar una nota.
- 2) El sistema recupera las notas existentes y muestra sus títulos.

#### Crear categoría (Flujo ppal)

- 1) Caso de uso se inicia cuando el usuario solicita crear categoría.
- 2) El usuario proporciona un título a la categoría.
- 3) El sistema guarda la categoría.

#### Modificar categoría (Flujo ppal)

- 1) Caso de uso se inicia cuando el usuario selecciona una categoría existente.
- 2) El usuario modifica el título de la categoría seleccionada.
- 3) El sistema guarda la categoría.

#### Eliminar categoría (Flujo ppal)

- 1) El caso de uso se inicia cuando se se selecciona una categoría existente y se solicita su borrado.
- 2) El sistema borra la nota.

#### Añadir categoría a nota (Flujo ppal)

- 1) El caso de uso se inicia cuando se está creando una nota.
- 2) El usuario selecciona una de las categorías que aparecen en la lista desplegable proporcionada.
- 3) El sistema guarda la nota con la categoría asignada.

#### Añadir categoría a nota (Flujo alternativo)

- 1) El caso de uso se inicia cuando se está modificando una nota.
- 2) El usuario selecciona una de las categorías que aparecen en la lista desplegable proporcionada.
- 3) El sistema guarda la nota con la categoría asignada.

#### Eliminar categoría de nota (Flujo ppal)

- 1) El caso de uso se inicia cuando se está modificando una nota.
- 2) El usuario selecciona la opción de “Sin categoría”, con lo que la nota no tendría ninguna categoría asignada.
- 3) El sistema guarda la nota modificada.

#### Ordenar notas por título (Flujo ppal)

- 1) El caso de uso se inicia cuando se selecciona la opción de filtrar notas.
- 2) El usuario selecciona la opción de “filtrar por nombre”.
- 3) El sistema recupera las notas existentes y muestra sus títulos ordenados alfabéticamente.

#### Mostrar las notas de una determinada Categoría (Flujo ppal)

- 1) El caso de uso se inicia cuando se selecciona la opción de filtrar notas.
- 2) El usuario selecciona la opción de “filtrar por categoría”.
- 3) El usuario selecciona una de las categorías de la lista desplegable proporcionada.
- 4) El sistema recupera las notas existentes y muestra todas aquellas cuya categoría asignada coincide con la categoría seleccionada en el paso anterior.

#### Ordenar notas por categoría (Flujo ppal)

- 1) El caso de uso se inicia cuando se selecciona la opción de filtrar notas.
- 2) El usuario selecciona la opción de “agrupar por categoría”.
- 3) El sistema recupera las notas existentes y muestra sus títulos y la categoría asignada a cada una de ellas, ordenadas alfabéticamente según el nombre de la categoría.

# Análisis

## Diagramas de clases de análisis

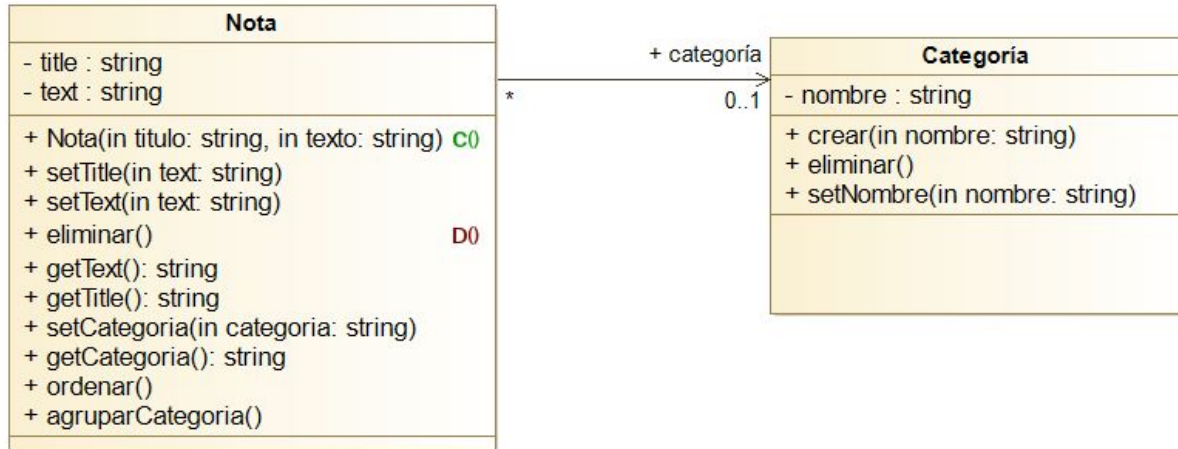


Diagrama de clases de análisis para el sistema descrito.

## Diagramas de secuencia para los Casos de Uso

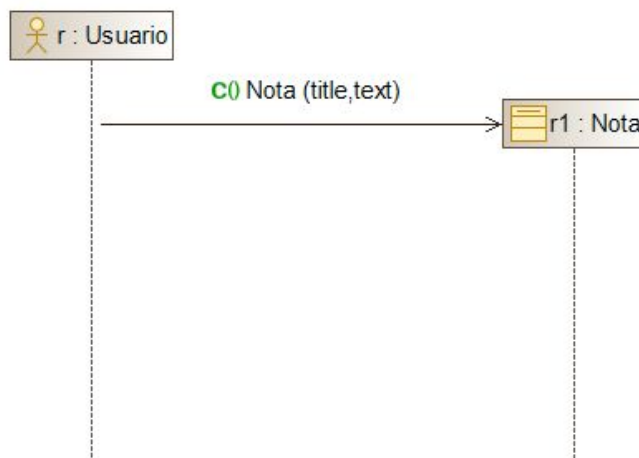


Diagrama de secuencia del CU Crear Nota

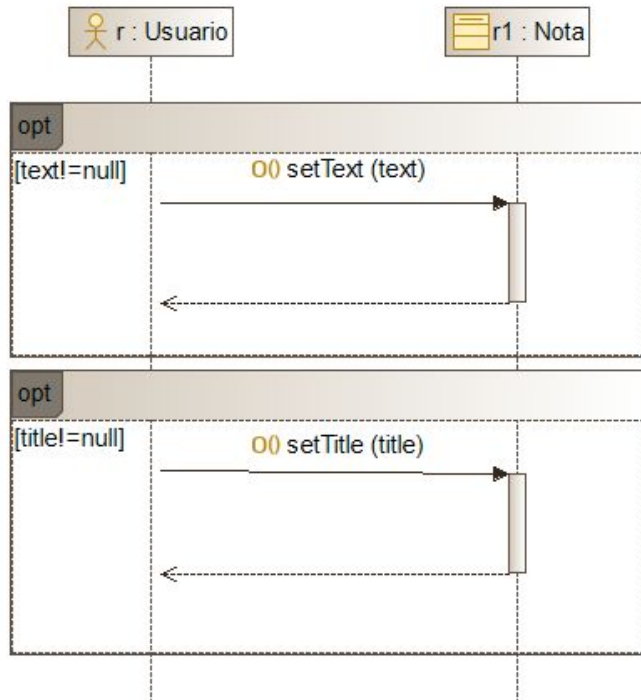


Diagrama de secuencia del CU Modificar Nota

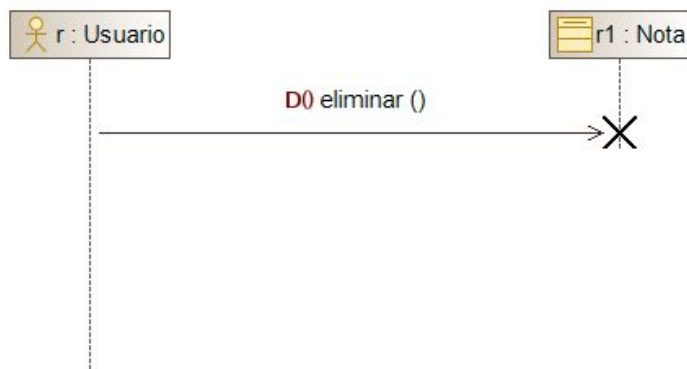


Diagrama de secuencia del CU Eliminar Nota

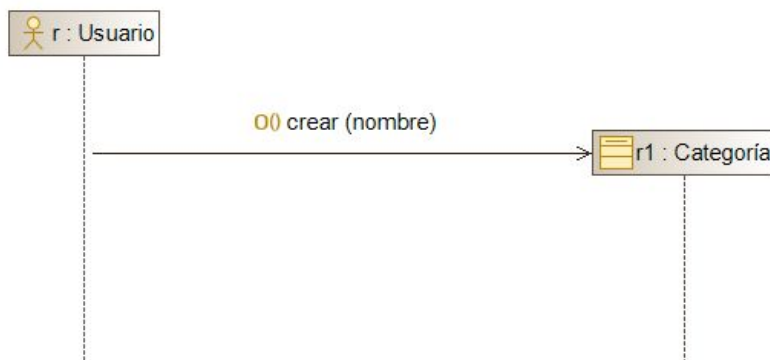


Diagrama de secuencia del CU Crear Categoría



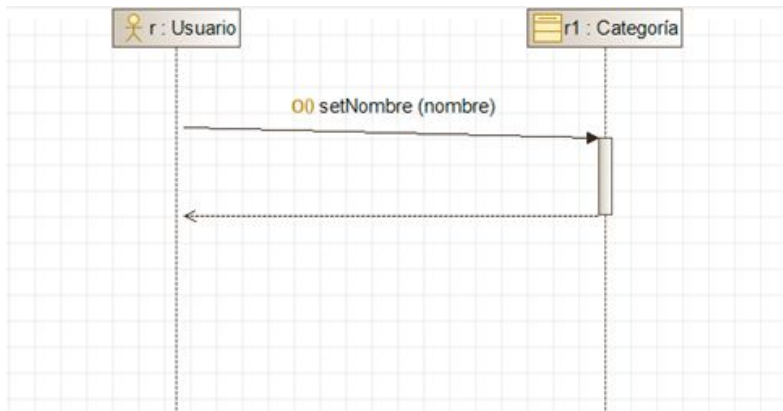


Diagrama de secuencia del CU Modificar Categoría

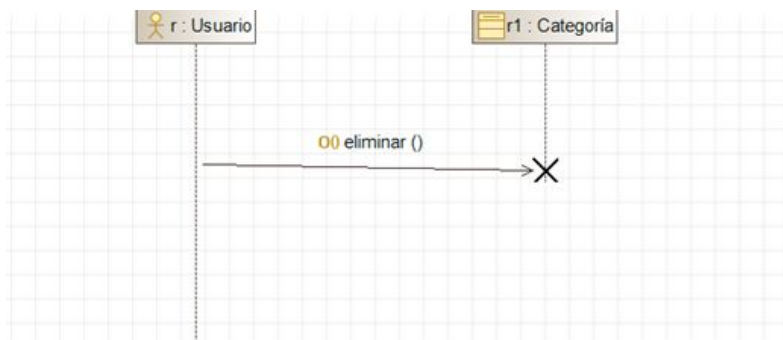


Diagrama de secuencia del CU Eliminar Categoría

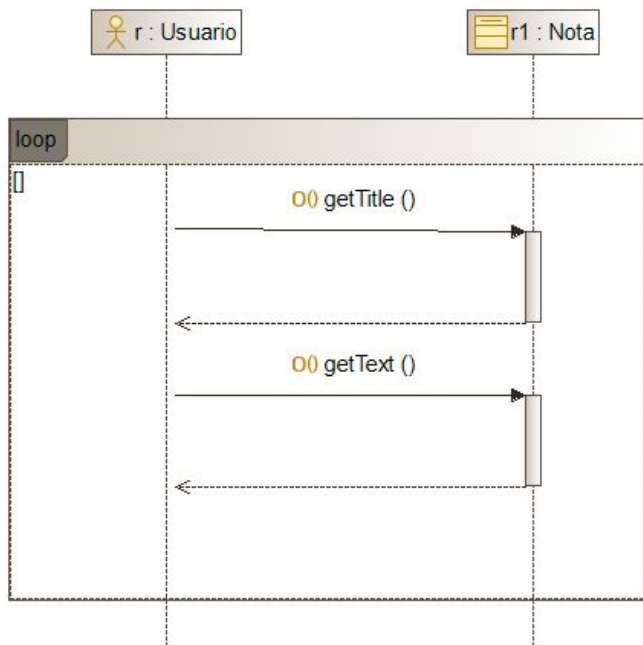


Diagrama de secuencia del CU Listar Notas

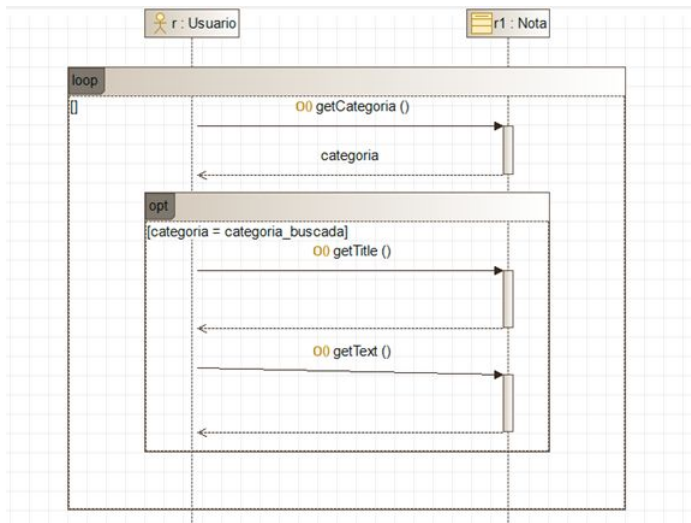


Diagrama de secuencia del CU Filtrar por Categoría

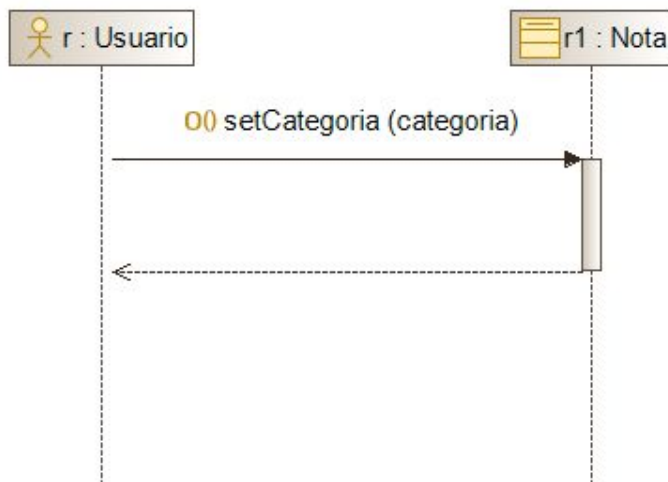


Diagrama de secuencia del CU Añadir categoría a nota

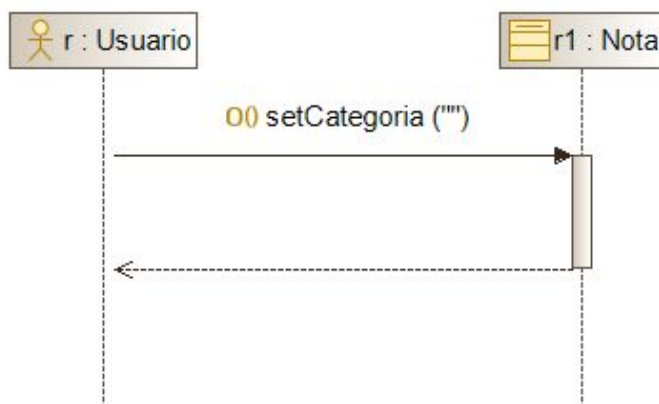


Diagrama de secuencia del CU Eliminar categoría a nota

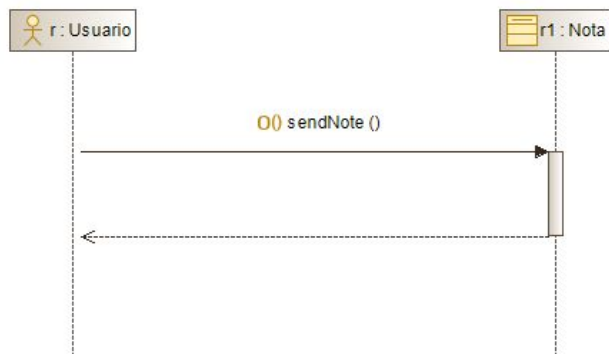


Diagrama de secuencia del CU enviar nota

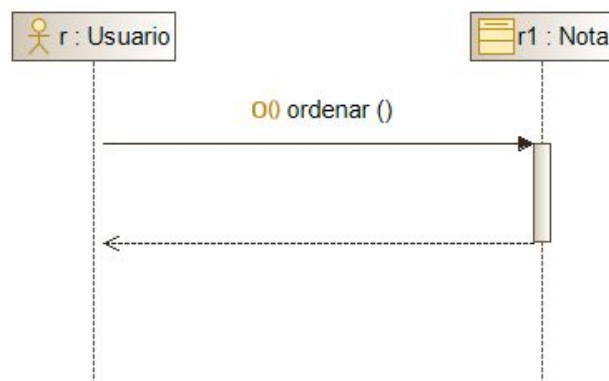


Diagrama de secuencia del CU ordenar por título

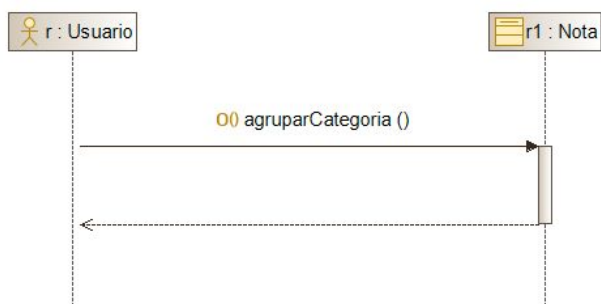
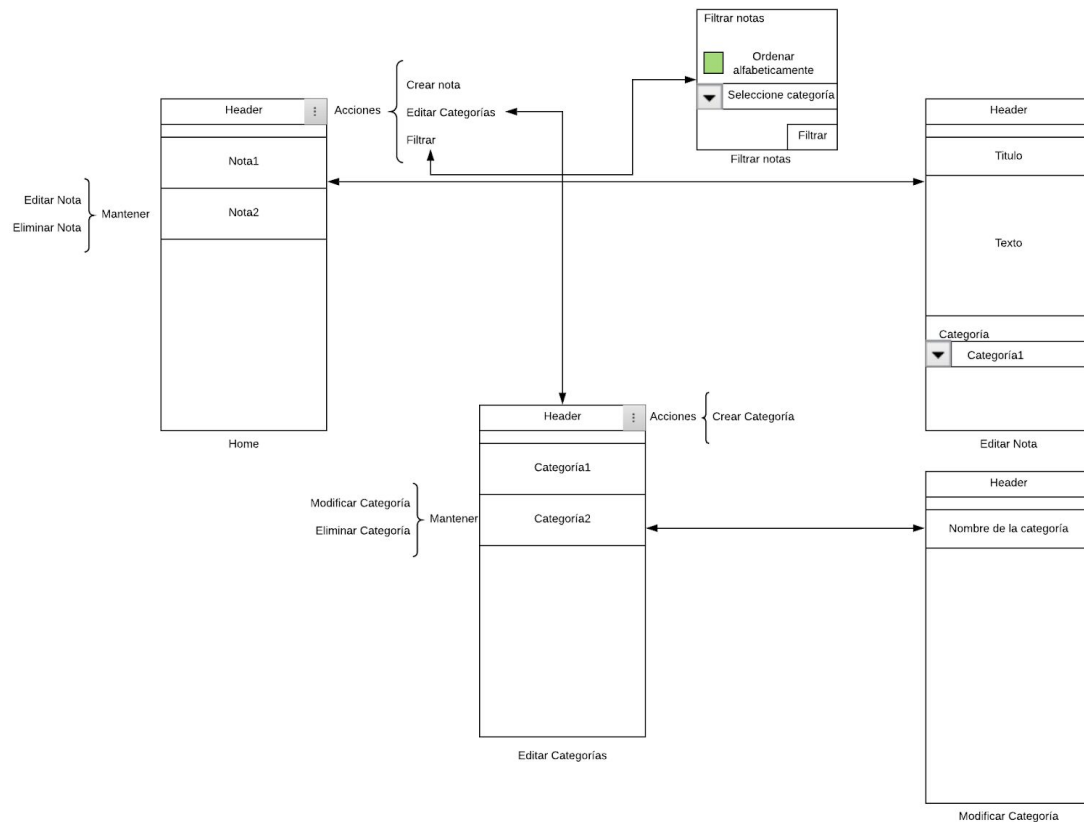


Diagrama de secuencia del CU agrupar por categoría

## Descripción de la interfaz de usuario

### Prototipos de las nuevas pantallas y mapa de navegación



### Mapa de navegación y prototipo de la aplicación

En la figura anterior se puede observar el mapa de navegación combinado con el prototipo de la aplicación. Cuando el usuario inicia la aplicación se muestran las notas que tuviera guardadas en la BD.

Para acceder a las distintas opciones de la aplicación deberá acceder al menú situado en la barra de título de la aplicación. Desde dicho menú podrá filtrar las notas existentes, ya sea por categoría (dada una categoría), ordenar por identificador, por orden alfabético o por identificador.

# Diseño del sistema

## Descomposición del sistema en diagramas de paquetes

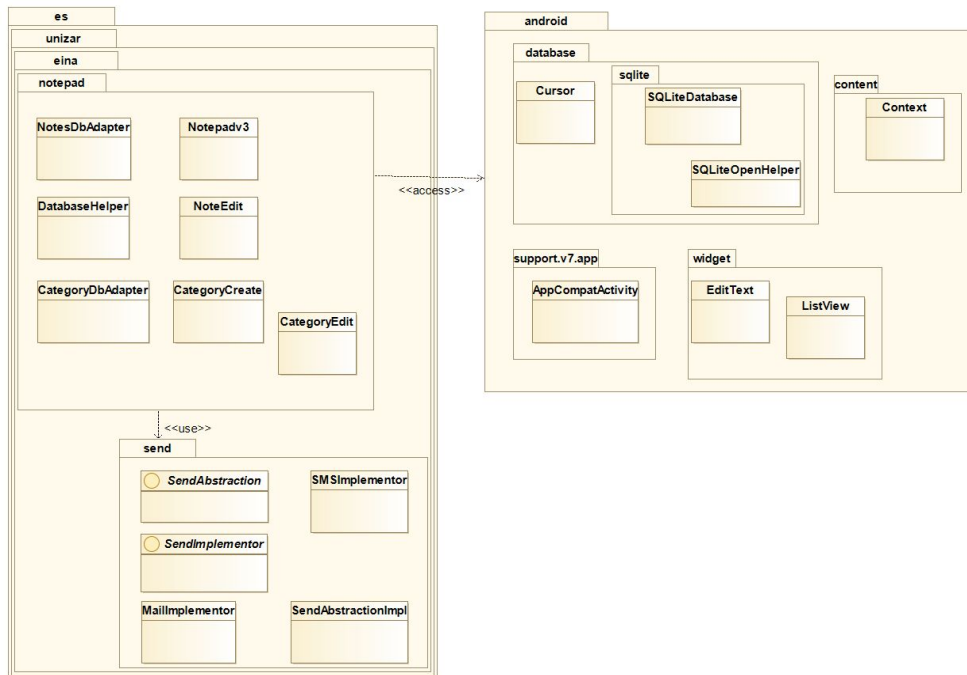
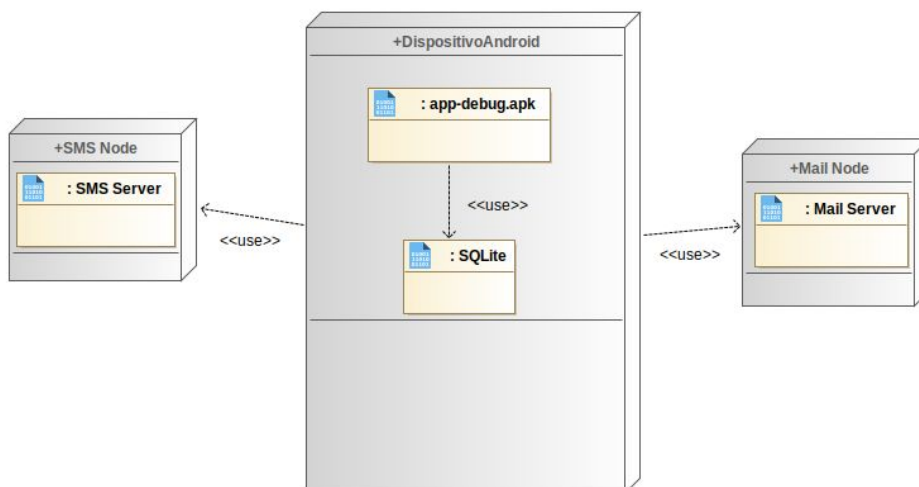


Diagrama de paquetes del sistema

## Diagrama de despliegue



## Distribución en componentes

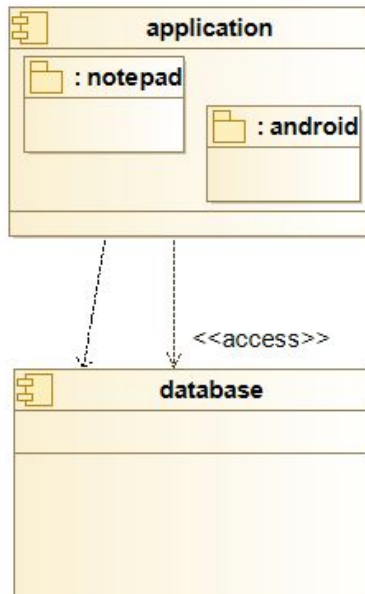
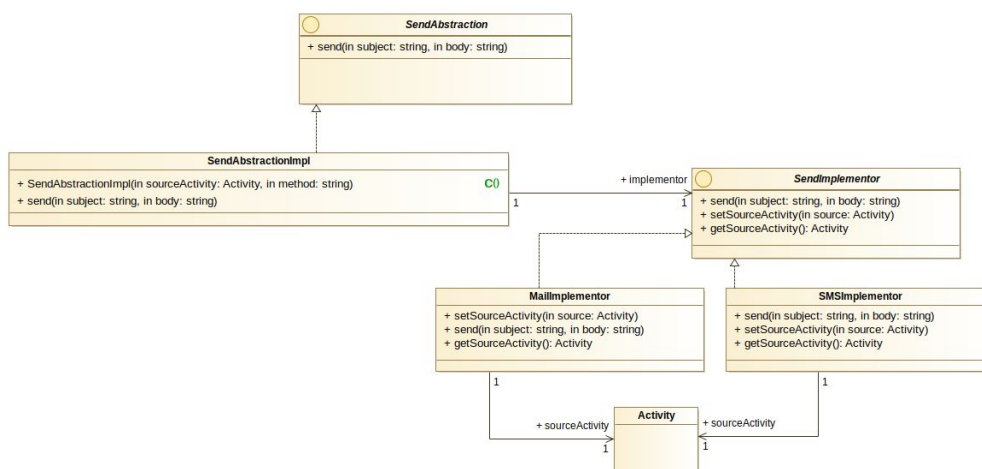


Diagrama de componentes del sistema.

## Diseño de objetos

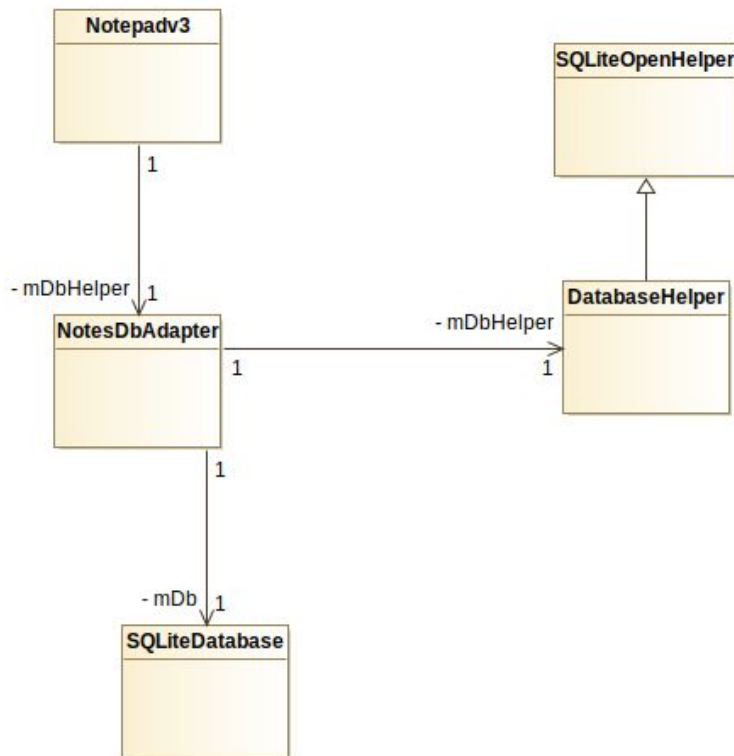
### Decisiones tomadas

### Patrón Bridge para envío de notas



Su uso es la elección de un *Implementor* que puede variar entre *SMSImplementor* y *MailImplementor* para realizar el envío de la nota correspondiente vía SMS o email.

## Patrón Adapter para el acceso a la Base de Datos



Se puede apreciar el patrón Adapter para facilitar al acceso a dos clases predefinidas en el *SDK* de Android. En este caso la clase *NotesDbAdapter* está encubriendo el acceso a la base de datos *SQLite* con la que trabaja la aplicación.

## Modelos estáticos

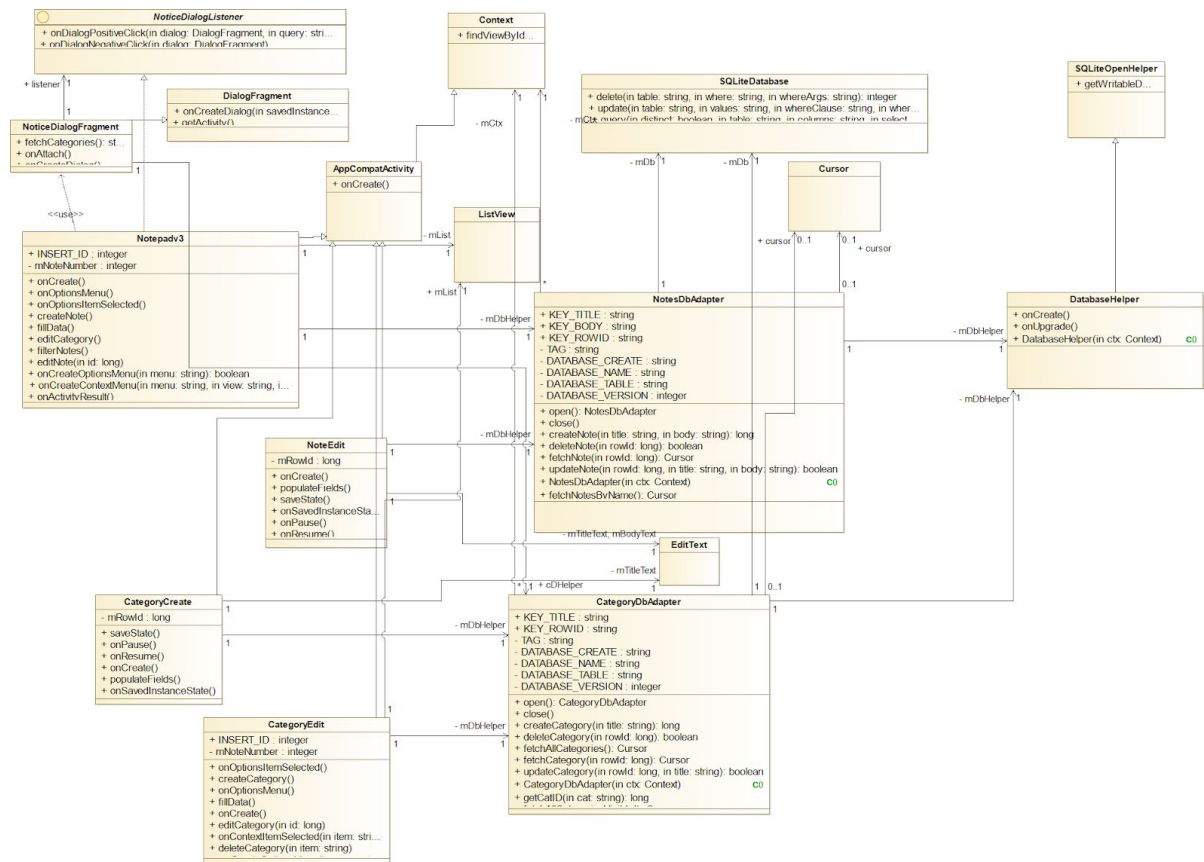
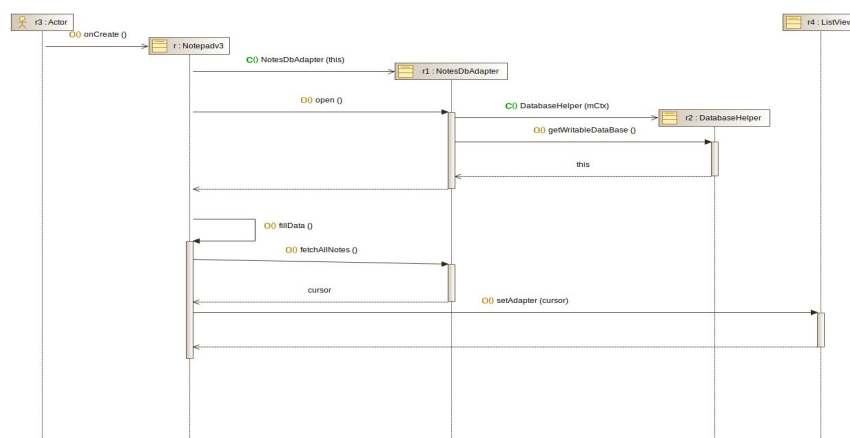


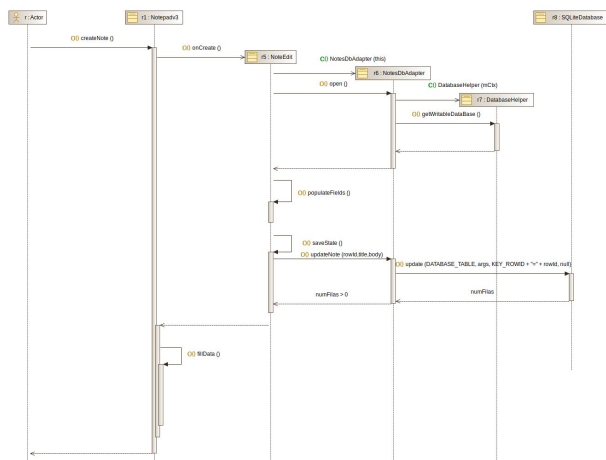
Diagrama de clases de diseño.

## Modelos dinámicos

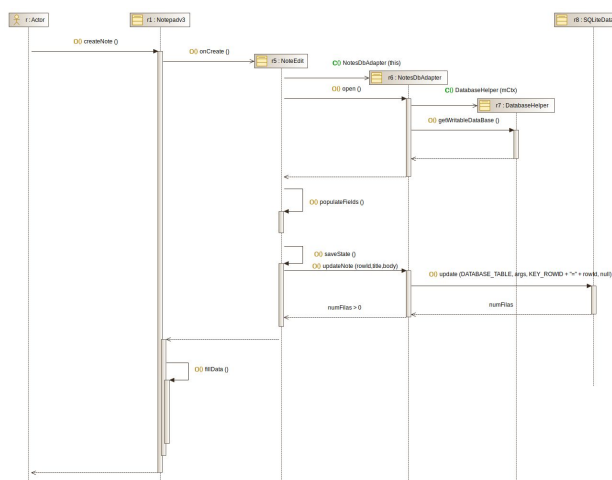


### Diagrama de secuencia de Iniciar aplicación

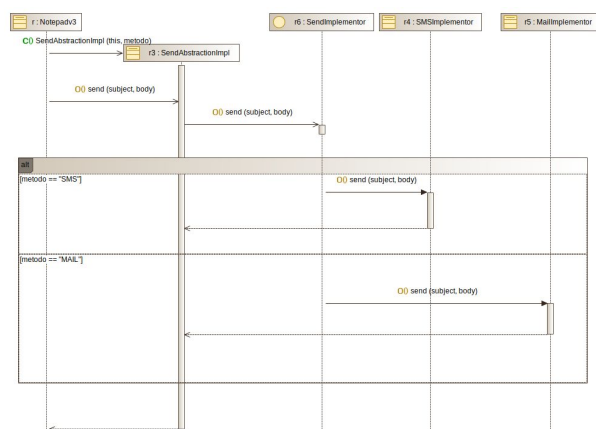




### Diagrama de secuencia Creación de nota



### Diagrama de secuencia Actualización de nota



### Diagrama de secuencia Envío de nota

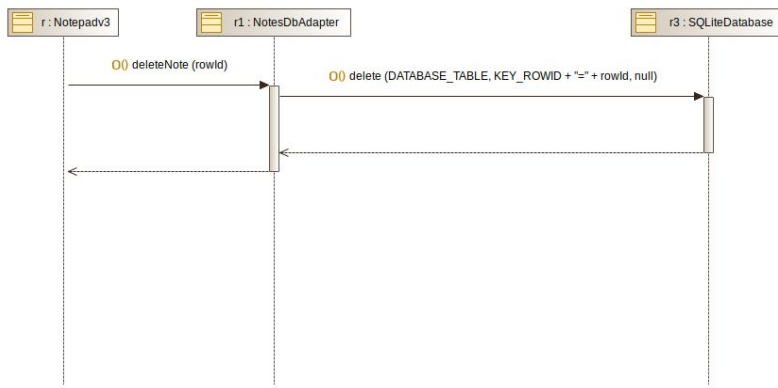


Diagrama de secuencia Borrar nota

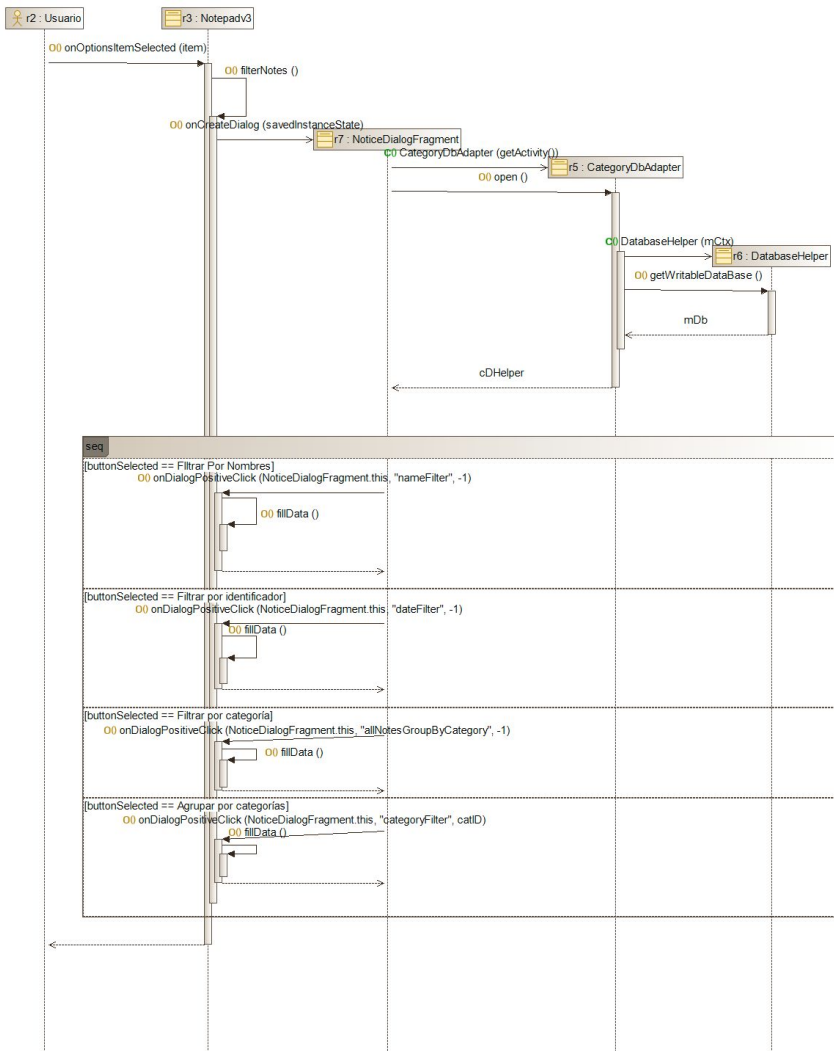


Diagrama de secuencia Aplicar Filtros (engloba agrupar por categoría, ordenar por identificador, ordenar por título y filtrar por categoría) y Listar Notas

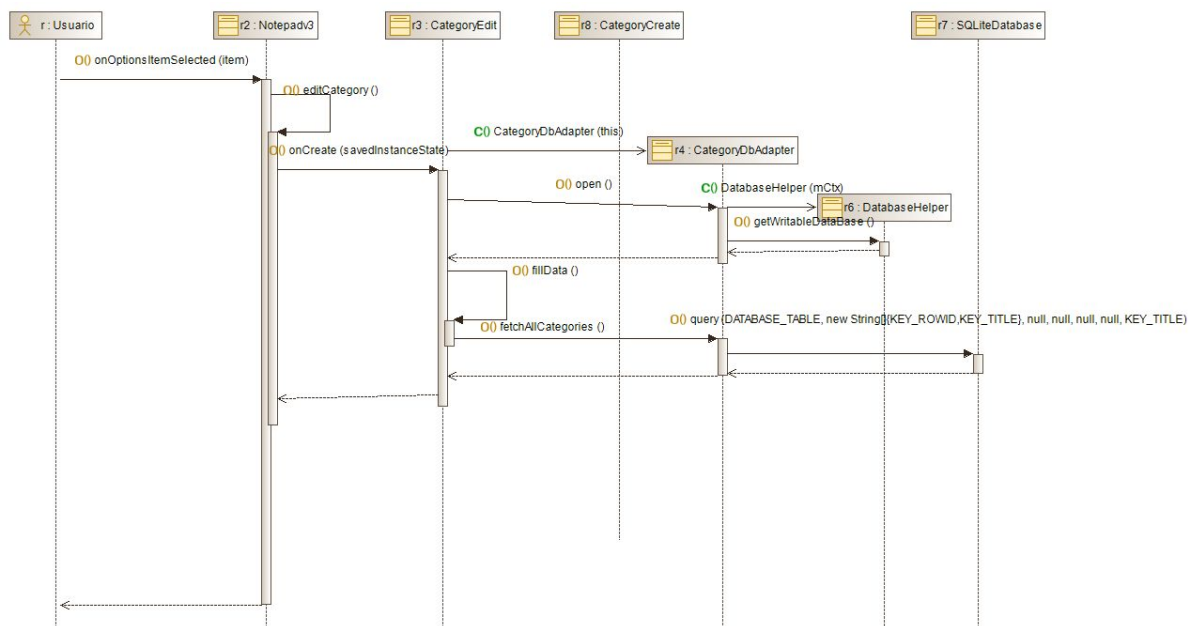


Diagrama de secuencia común de acceso a la actividad “editCategory”

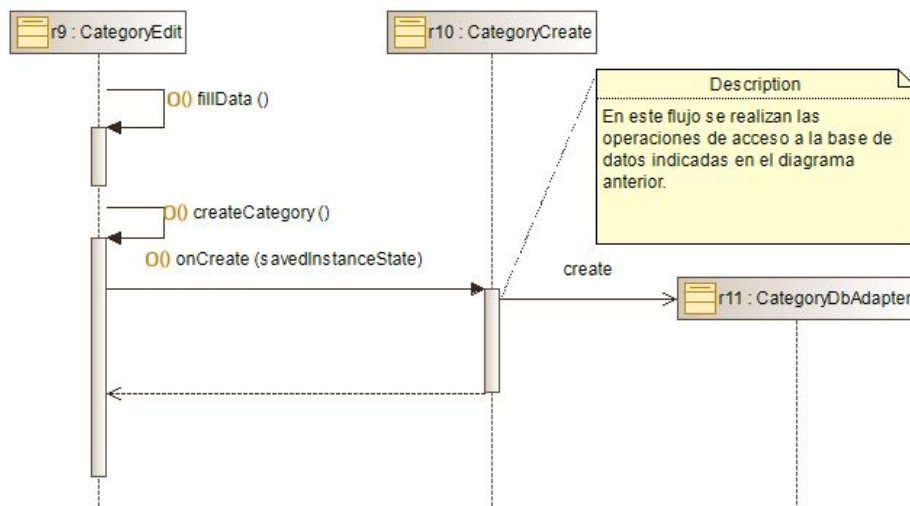


Diagrama de secuencia Crear Categoría

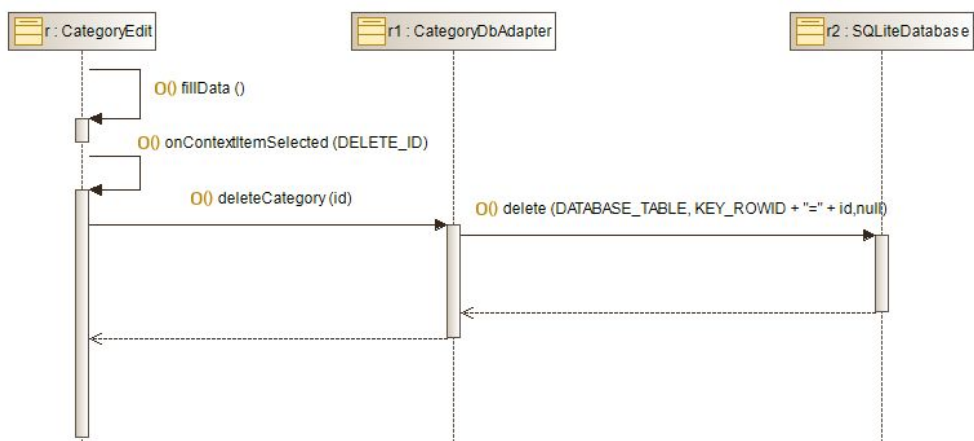


Diagrama de secuencia Eliminar Categoría

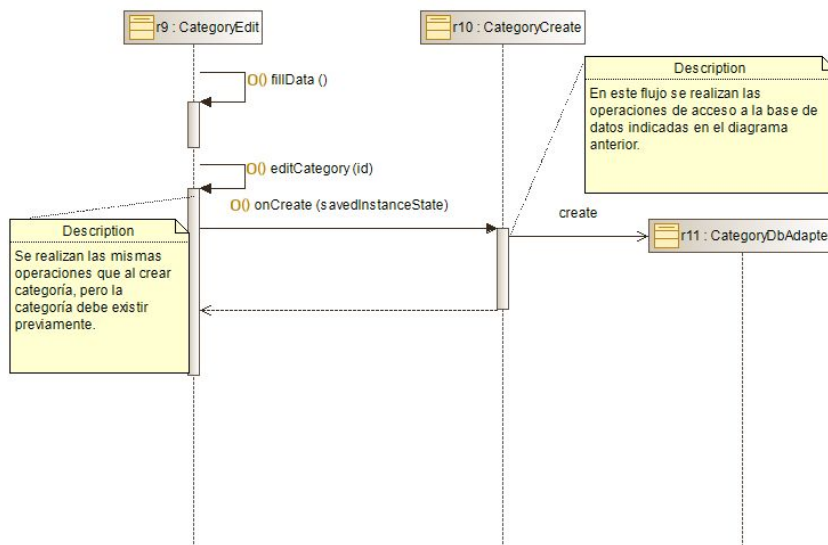


Diagrama de secuencia Modificar categoría

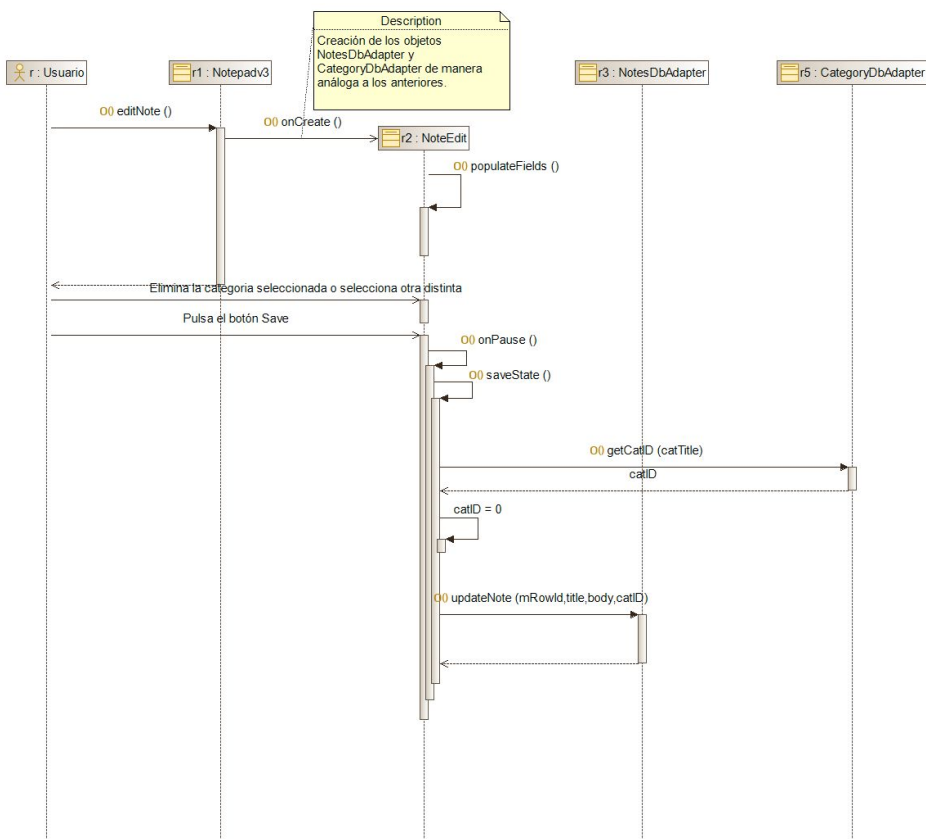


Diagrama de secuencia Añadir categoría a nota

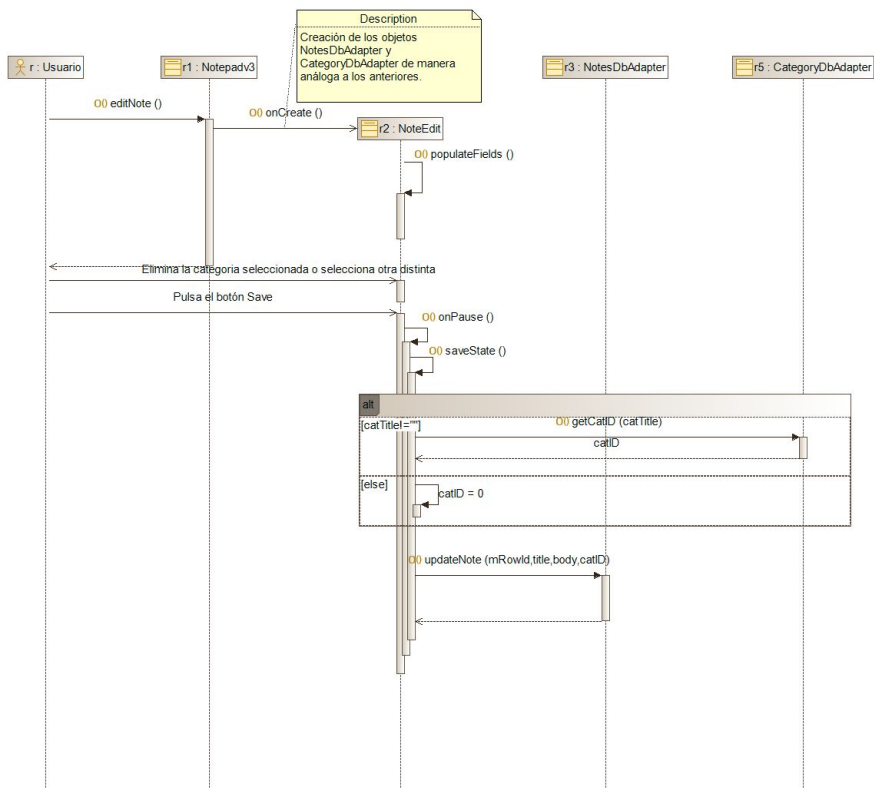


Diagrama de secuencia Eliminar categoría a nota

## Modelo E-R de la BBDD

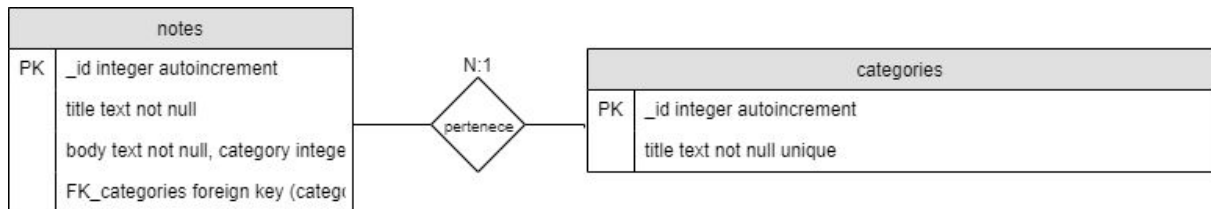


Diagrama E-R de la BBDD SQLite utilizada en la aplicación de notas

En una primera versión de la aplicación, existían dos bases de datos diferentes, una para notas y otra para categorías. Esta decisión se tomó debido a que aparecieron algunos problemas a la hora de intentar introducir ambas tablas en la misma base.

La versión actual consiste de una única base de datos, “*dataNotes*”, que posee dos tablas, una para Notas y otra para categorías. La tabla Notes posee un campo, “category”, que referencia al campo “\_id” de la tabla “categories”.

# Pruebas

## Pruebas unitarias y de caja negra

Clases de equivalencia: updateNote(long rowId, String title, String body)

Entrada	Clases de equivalencia válidas	Clases de equivalencia inválidas
title	1) title != Null 2) title.length > 0	5) title == Null 6) title.length() == 0
body	3) body != null	7) body == Null
rowId	4) rowId > 0	8) rowId <= 0

## Resultado de ejecución de los tests

```
2020-01-08 10:19:43.894 21207-21207/es.unizar.eina.notepadv1 D/TEST: ---- Inicializando test ----
2020-01-08 10:19:43.898 21207-21207/es.unizar.eina.notepadv1 D/TEST: createNote() con parámetros (Título prueba, Cuerpo prueba) finalizado correctamente con id 1
2020-01-08 10:19:43.899 21207-21207/es.unizar.eina.notepadv1 D/TEST: createNote() con parámetros (null, Cuerpo prueba) ha fallado. Causa: java.lang.NullPointerException: Attempt to invoke virtual method 'int java.lang.String.length()' on a null object reference
2020-01-08 10:19:43.899 21207-21207/es.unizar.eina.notepadv1 D/TEST: createNote() con parámetros ("", Cuerpo prueba) finalizado correctamente con id -1
2020-01-08 10:19:43.905 21207-21207/es.unizar.eina.notepadv1 D/TEST: createNote() con parámetros (Título prueba', null) ha fallado. Causa: android.database.sqlite.SQLiteConstraintException: NOT NULL constraint failed: notes.body (code 1299 SQLITE_CONSTRAINT_NOTNULL)
2020-01-08 10:19:43.908 21207-21207/es.unizar.eina.notepadv1 D/TEST: updateNote() con parámetros (1 ,Título prueba, Cuerpo prueba) finalizado correctamente con resultado true
2020-01-08 10:19:43.909 21207-21207/es.unizar.eina.notepadv1 D/TEST: updateNote() con parámetros (1, "", Cuerpo prueba) finalizado correctamente con resultado false
2020-01-08 10:19:43.909 21207-21207/es.unizar.eina.notepadv1 D/TEST: updateNote() con parámetros (1, null, Cuerpo prueba) ha fallado. Causa: java.lang.NullPointerException: Attempt to invoke virtual method 'int java.lang.String.length()' on a null object reference
2020-01-08 10:19:43.911 21207-21207/es.unizar.eina.notepadv1 D/TEST: updateNote() con parámetros (1, Título prueba, null) ha fallado. Causa: android.database.sqlite.SQLiteConstraintException: NOT NULL constraint failed: notes.body (code 1299 SQLITE_CONSTRAINT_NOTNULL)
2020-01-08 10:19:43.913 21207-21207/es.unizar.eina.notepadv1 D/TEST: updateNote() con parámetros (0 ,Título prueba, Cuerpo prueba) finalizado correctamente con resultado false
2020-01-08 10:19:43.915 21207-21207/es.unizar.eina.notepadv1 D/TEST: deleteNote() con parámetro (1) finalizado correctamente con resultado true
2020-01-08 10:19:43.916 21207-21207/es.unizar.eina.notepadv1 D/TEST: deleteNote() con parámetro (0) finalizado correctamente con resultado false
```

## Prueba de volumen

Para la realización de la prueba de carga, se ha realizado un método encargado de introducir 1000 notas, con un título compuesto de un prefijo común “Nota carga” + Num. nota. La captura adjunta muestra las últimas notas de la lista.

A pesar de haber 1000 notas almacenadas, no ha aparecido ningún problema en la aplicación.

Al modificar cualquier nota, cuando se vuelve a la pantalla principal, se muestra la lista de notas a partir de la editada.

Notepadv3
Nota carga 972
Nota carga 973
Nota carga 974
Nota carga 975
Nota carga 976
Nota carga 977
Nota carga 978
Nota carga 979
Nota carga 98
Nota carga 980
Nota carga 981
Nota carga 982
Nota carga 983
Nota carga 984
Nota carga 985
Nota carga 986
Nota carga 987
Nota carga 988
Nota carga 989
Nota carga 99
Nota carga 990
Nota carga 991
Nota carga 992
Nota carga 993
Nota carga 994
Nota carga 995
Nota carga 996
Nota carga 997
Nota carga 998
Nota carga 999

## Prueba de sobrecarga

### Pruebas ejecutadas en el emulador

Para la realización de las pruebas de sobrecarga sobre nuestra aplicación, se ha desarrollado un método encargado de crear notas de tamaño creciente hasta que se detecte un fallo. Cada una de las notas contiene  $i * 10$  caracteres, siendo  $i$  el número de nota. El resultado de la ejecución ha sido:

```
2020-01-08 11:23:27.254 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103180 caracteres.
2020-01-08 11:23:27.320 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103190 caracteres.
2020-01-08 11:23:27.384 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103200 caracteres.
2020-01-08 11:23:27.445 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103210 caracteres.
2020-01-08 11:23:27.506 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103220 caracteres.
2020-01-08 11:23:27.568 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103230 caracteres.
2020-01-08 11:23:27.633 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103240 caracteres.
2020-01-08 11:23:27.690 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103250 caracteres.
2020-01-08 11:23:27.760 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103260 caracteres.
2020-01-08 11:23:27.817 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103270 caracteres.
2020-01-08 11:23:27.878 4549-4549/es.unizar.elna.notepadv1 D/TEST: Almacenada nota de 103280 caracteres.
2020-01-08 11:23:27.933 4549-4549/es.unizar.elna.notepadv1 D/TEST: Fallo realizando test de carga con 103290 caracteres. Causa: android.database.sqlite.SQLiteFullException: database or disk is full
(code 13)
```

Se puede observar que al introducir una nota de 103290 caracteres, es decir, 10329 notas, la aplicación ha lanzado una excepción debido a que la base de datos se ha llenado por completo (sqlite.SQLiteFullException).

### Pruebas ejecutadas con monkey

Para la ejecución de eventos aleatorios en la aplicación, se ha utilizado la herramienta monkey, mediante el comando: “adb shell monkey -p es.unizar.elna.notepadv1 -v2 500”

Este comando genera 500 eventos aleatorios de la aplicación Notepadv1.

El resultado de la ejecución ha sido:

```
lamborg@lamborg99-laptop:~/Documents/Universidad/3o/Ingenieria_Software/Practicas/Ingenieria_sw/pres adb shell monkey -p es.unizar.elna.notepadv1 -v2 500
bash arg: -p
bash arg: es.unizar.elna.notepadv1
bash arg: -v2
bash arg: 500
args: [-p es.unizar.elna.notepadv1, -v2, 500]
arg: -p
arg: es.unizar.elna.notepadv1
arg: -v2
arg: 500
data="es.unizar.elna.notepadv1"
Monkey: seed=1578796894 count=500
AllowPackage: es.unizar.elna.notepadv1
IncludeCategory: android.intent.category.LAUNCHER
IncludeCategory: android.intent.category.MONKEY
// Event percentages:
// 0: 15.0%
// 1: 10.0%
// 2: 2.0%
// 3: 15.0%
// 4: -0.0%
// 5: -0.0%
// 6: 25.0%
// 7: 15.0%
// 8: 2.0%
// 9: 2.0%
// 10: 1.0%
// 11: 13.0%
Switch: #Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;launchFlags=0x10200000;component=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3;end
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
(Sending Touch (ACTION_DOWN): 0:(815.0,323.0)
Sending Touch (ACTION_UP): 0:(813.075,335.77542)
Sending Trackball (ACTION_MOVE): 0:(-3.0,3.0)
Sending Trackball (ACTION_MOVE): 0:(0.0,0.0)
Sending Trackball (ACTION_MOVE): 0:(4.0,-2.0)
Switch: #Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;launchFlags=0x10200000;component=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3;end
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
(Sending Touch (ACTION_UP): 0:(913.86224,1689.457)
Switch: #Intent;action=android.intent.action.MAIN;category=android.intent.category.LAUNCHER;launchFlags=0x10200000;component=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3;end
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
// Allowing start of Intent ( act=android.intent.action.MAIN cat=[android.intent.category.LAUNCHER] cmp=es.unizar.elna.notepadv1/es.unizar.elna.notepadv3.Notepadv3 ) in package es.unizar.elna.notepadv1
(Sending Touch (ACTION_DOWN): 0:(955.0,1698.0)
(Sending Touch (ACTION_UP): 0:(996.217,940.39853)
Sending Trackball (ACTION_MOVE): 0:(-5.0,-5.0)
(Sending Touch (ACTION_DOWN): 0:(918.0,1241.0)
(Sending Touch (ACTION_UP): 0:(922.3041,1244.155)
(Sending Touch (ACTION_DOWN): 0:(52.0,791.0)
(Sending Touch (ACTION_UP): 0:(71.1515,801.1527)
(Sending Touch (ACTION_DOWN): 0:(925.0,800.0)
(Sending Touch (ACTION_UP): 0:(920.5445,813.7369)
(Sending Touch (ACTION_DOWN): 0:(3015.0,1255.0)
(Sending Touch (ACTION_UP): 0:(1076.8625,1243.0793)
Sending Trackball (ACTION_MOVE): 0:(-4.0,-1.0)
// [calendar_time:2020-01-08 09:50:51.685 system_uptime:3884866344]
// Sending event #500
Sending Trackball (ACTION_MOVE): 0:(3.0,0.0)
Sending Touch (ACTION_DOWN): 0:(183.0,1969.0)
Sending Touch (ACTION_UP): 0:(124.54236,1964.7037)
Sending Trackball (ACTION_MOVE): 0:(2.0,-2.0)
Sending Touch (ACTION_DOWN): 0:(610.0,146.0)
Events Injected: 500
Sending rotation degree=0, persist=false
Dropped: keys=0 pointers=0 trackballs=0 flaps=2 rotations=0
## Network stats: elapsed time=745ms (0ms mobile, 0ms wifi, 745ms not connected)
// Monkey finished
```

## Resultados y conclusiones

Como resultado del trabajo anterior se ha obtenido una aplicación para dispositivos Android con la cual podemos gestionar un conjunto de notas, ofreciendo la posibilidad de agruparlas en categorías. Las categorías están compuestas por un identificador único y un nombre.

La aplicación permite modificar el orden de visualización de las notas existentes, siendo posible ordenarlas: por título, por el identificador interno de cada nota o mostrar las notas de una determinada categoría.

Un punto fuerte de la aplicación es, de cara al usuario, la consistencia en cuanto a los controles que nos encontramos dentro de la aplicación, ya que el usuario puede aprender que si manteniendo pulsado sobre una nota puede desplegar un menú para gestionar la misma podrá realizar la misma acción sobre cada categoría, obteniendo un comportamiento similar al de las notas.

Otro punto fuerte sobre la aplicación es el amplio abanico de posibilidades de filtrado o agrupación de las que dispone la aplicación, ya que brinda la posibilidad de agrupar por categorías, pudiendo mostrar todas las notas pertenecientes a una categoría de forma contigua en el listado, también se puede filtrar por una categoría determinada, mostrando solo las notas de dicha categoría. También dispone de opciones de ordenación, pudiendo ordenarse por el identificador interno de la propia o por el título asignado a la misma.

Un punto débil de la aplicación se corresponde con el apartado estético, la presentación podría mejorar utilizando otro tipo de fuentes, otros colores, cambiando el icono de la aplicación...

Un punto de mejora de la aplicación podría ser el formato de envío de una nota a través de correo electrónico, añadiendo un formato HTML al email.

Otro posible punto de mejora a futuro sería la sincronización de las notas y categorías entre los dispositivos de un mismo usuario, permitiendo a un usuario acceder y modificar sus notas en distintos dispositivos.