

TRABAJO LS.

APLICACION

NOTEPAD

FERNANDO ALIAGA RAMON – 610610

CRISTIAN SIMON MORENO - 611487

INDICE

Introduccion

Requisitos

- Catalogo de requisitos
- Casos de uso y descripción de escenarios

Analisis

- Modelos estaticos
- Modelos dinamicos
- Descripcion de la interfaz de usuario

Diseño del sistema

- Diagrama de clases
- Diagrama de paquetes
- Diagrama de componentes y de despliegue
- Diagramas de modelos estaticos y dinamicos

Diseño de objetos

- Descripcion de decisiones de diseño
- Diagramas correspondientes a los modelos estaticos y dinamicos
- Modelo relacional de la base de datos

Pruebas

Bibliografia

INTRODUCCION:

Los objetivos del trabajo son permitir la existencia de categorías (creación, modificación, eliminación, ordenación y listado) dentro de la aplicación de notas realizada en las prácticas de la asignatura.

Para su realización haremos uso de los diferentes diagramas y esquemas que nos permitirán analizar los requisitos, el análisis y el diseño.

REQUISITOS

CATALOGO DE REQUISITOS

➤ FUNCIONALES

A parte de los requisitos funcionales vistos en la practicas anteriores, tendremos:

- Permitir la creación de notas.
- Permitir la modificación de una nota.
- Permitir la eliminación de una nota.
- Permitir el envío de una nota previamente creada por email.
- Permitir la creación de categorías.
- Permitir la modificación de una categoría.
- Permitir la eliminación de categorías, si hay notas con dicha categoría se conservaran sin una categoría asignada.
- Permitir asignar a cada nota una sola categoría previamente creada.
- Permitir modificar la categoría previamente asignada a una nota.
- Permitir eliminar la categoría previamente asignada a una nota.
- Permitir ordenar el listado de notas por categoría o por título.
- Permitir que el listado de notas muestre todas las notas o solo las de una determinada categoría.

➤ NO FUNCIONALES

- Portabilidad:
 - La aplicación deberá visualizarse y funcionar correctamente en cualquier dispositivo android.
- Interfaz de usuario:
 - La interfaz sigue el estándar de android.
- Soporte:
 - El sistema debe ser instalable por los usuarios.
- Recursos:
 - El sistema no será estable con mas de 1000 notas en la base de datos.
- Operatividad:
 - Permitir la existencia de categorías de notas. Una categoría consiste, desde el punto de vista del usuario, en una o varias palabras

compuestas por letras y/o números y separadas entre sí por un espacio en blanco como máximo.

CASOS DE USO

- Crear nota:

Principal

1. El caso de uso comienza cuando el usuario desea crear una nota.
2. El usuario da nombre, cuerpo a la nota, así como poder asignarle una categoría previamente creada.
3. El sistema guarda la nota
4. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

1. El usuario puede cancelar la creación de la nota en cualquier momento

- Modificar nota:

Principal

1. El caso de uso comienza cuando el usuario selecciona una nota.
2. La aplicación muestra el nombre, cuerpo y categoría, si tiene, de la nota.
3. El usuario modifica el nombre, cuerpo o categoría de la nota.
4. El sistema guarda la nota
5. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

2. El usuario puede cancelar la edición de la nota en cualquier momento

- Guardar nota:

Principal

1. El usuario solicita guardar una nota.
2. El sistema guarda la nota
3. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

3. El usuario puede cancelar el guardado de la nota en cualquier momento

- Borrar nota:

Principal

1. El caso de uso comienza cuando el usuario selecciona una nota y solicita borrar dicha nota.
2. La aplicación muestra el nombre, cuerpo y categoría, si tiene, de la nota.
3. El usuario solicita el borrado de la nota.
4. El sistema borra la nota
5. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

4. El usuario puede cancelar el borrado de la nota en cualquier momento

- Mandar por mail nota:

Principal

1. El caso de uso comienza cuando el usuario selecciona una nota y solicita mandar por mail dicha nota.
2. La aplicación solicita los datos tales como email de origen, destino, contraseña, password, host, puertos y los campos de dicha nota (titulo, cuerpo y si tiene, la categoría).
3. El usuario modifica dichos datos y le da a mandar.
4. El sistema manda la nota
5. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

5. El usuario puede cancelar el envio por mail de la nota en cualquier momento

- Crear categoria

Principal

1. El caso de uso comienza cuando selecciona gestionar categorías.
2. El usuario selecciona añadir categoría.
3. El usuario da nombre a la categoría.
4. El sistema guarda la categoria
5. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

6. El usuario puede cancelar la creación de la categoria en cualquier momento

- Modificar categoria:

Principal

1. El caso de uso comienza cuando selecciona gestionar categorías.
2. El usuario selecciona una categoría.
3. El usuario modifica el nombre a la categoría.
4. El sistema guarda la categoría.
5. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

1. El usuario puede cancelar la edición de la categoria en cualquier momento

- Gestionar categoria:

Principal

1. El usuario solicita gestionar categorias.
2. El sistema gestiona la categoría.
3. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

1. El usuario puede cancelar la gestión de categorias en cualquier momento

- Borrar categorías:

Principal

1. El caso de uso comienza cuando selecciona gestionar categorías.
2. El usuario selecciona una categoría.
3. La aplicación muestra el nombre de la categoría.
4. El usuario solicita el borrado de la categoría.
5. El sistema borra la categoría.
6. La aplicación vuelve al estado inicial y termina el caso de uso.

Alternativo

1. El usuario puede cancelar el borrado de categorías en cualquier momento

- Listar todas:

Principal

1. El caso de uso comienza cuando selecciona listar.
2. El usuario selecciona las opciones de listado.
3. La aplicación muestra el listado de notas de acuerdo con las especificaciones del usuario (listar todas).
4. La aplicación vuelve al estado inicial con el listado actualizado y termina el caso de uso.

Alternativo

1. El usuario puede cancelar el listado de las notas en cualquier momento

- Listar por categorías:

Principal

1. El caso de uso comienza cuando selecciona listar.
2. El usuario selecciona las opciones de listado.
3. La aplicación muestra el listado de notas de acuerdo con las especificaciones del usuario (listar por categorías).
4. La aplicación vuelve al estado inicial con el listado actualizado y termina el caso de uso.

Alternativo

2. El usuario puede cancelar la creación de la nota en cualquier momento

- Listar:

Principal

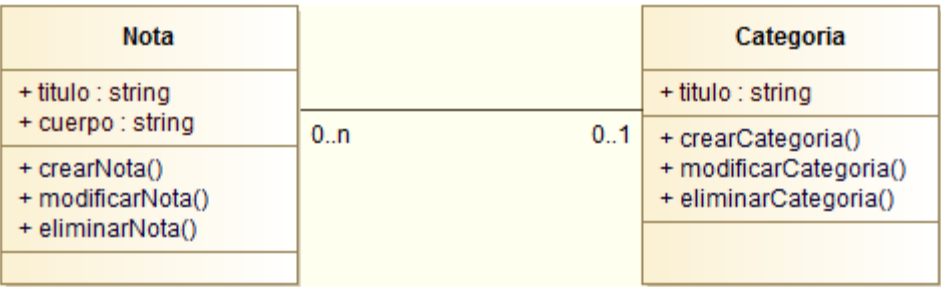
1. El usuario solicita listar las notas modificando las opciones de listado.
2. El sistema lista las nota
3. La aplicación vuelve al estado inicial con el listado actualizado y termina el caso de uso.

Alternativo

7. El usuario puede cancelar el listado de las notas en cualquier momento

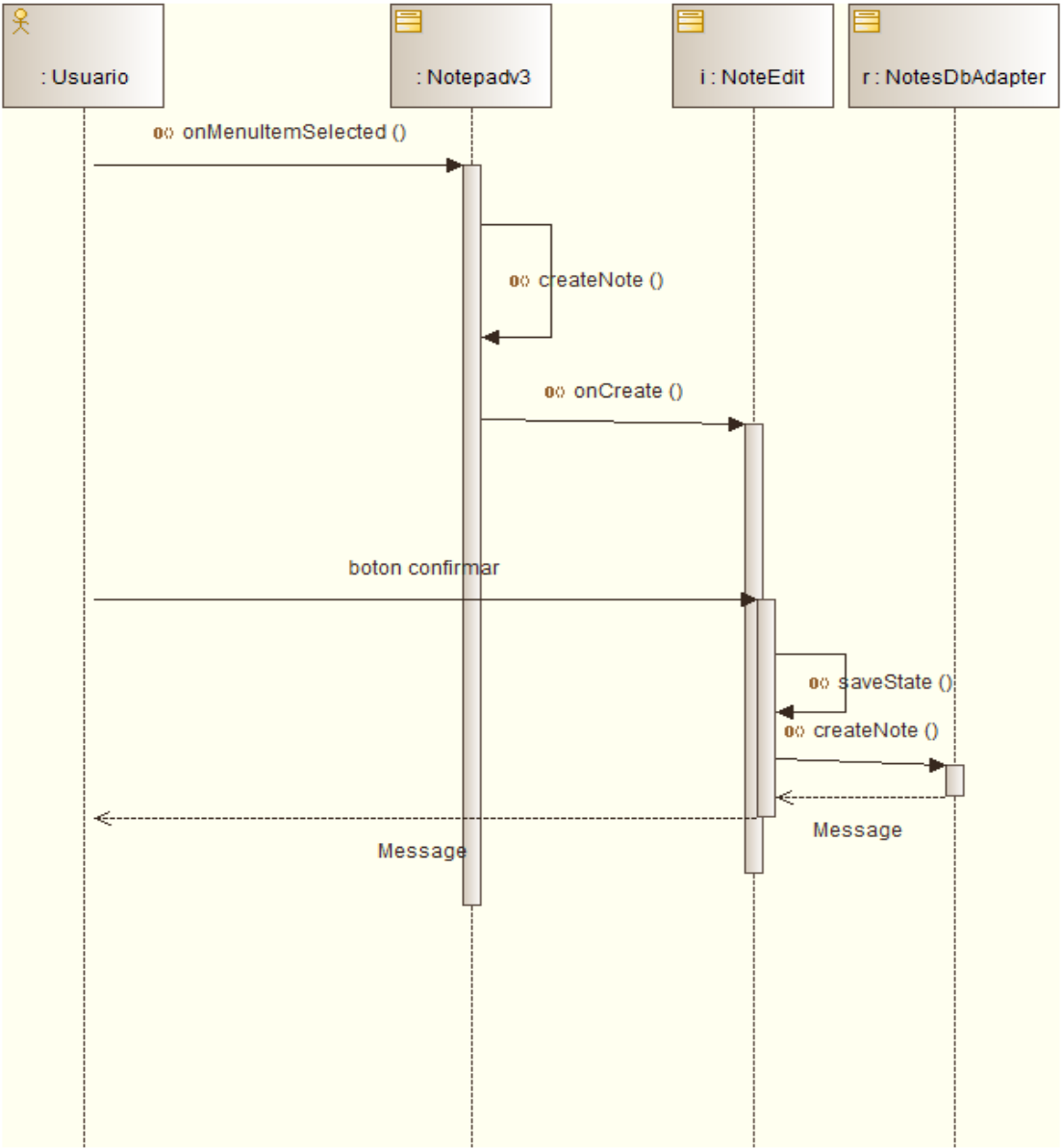
ANALISIS

DIAGRAMA DE CLASES

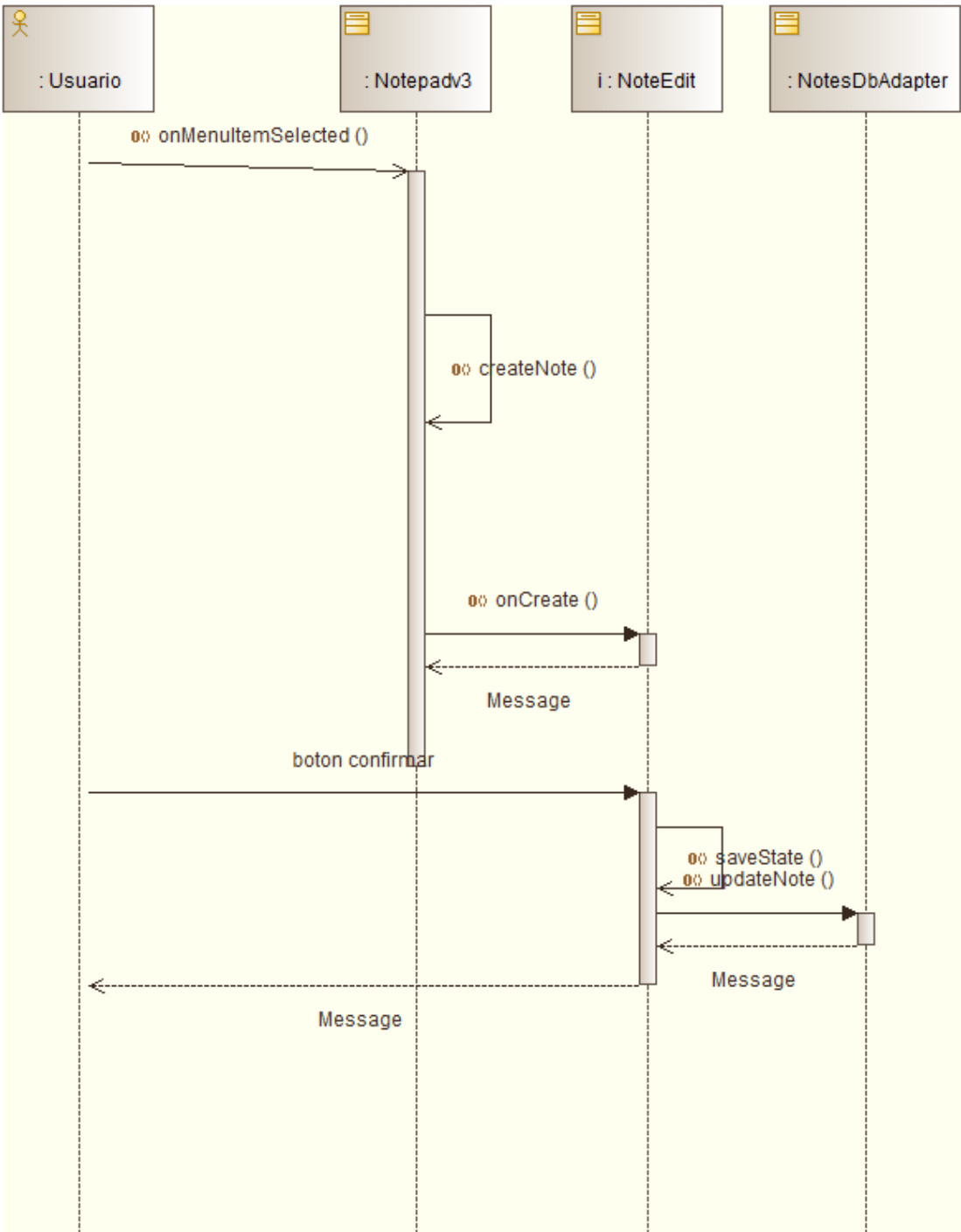


DIAGRAMAS DE SECUENCIA

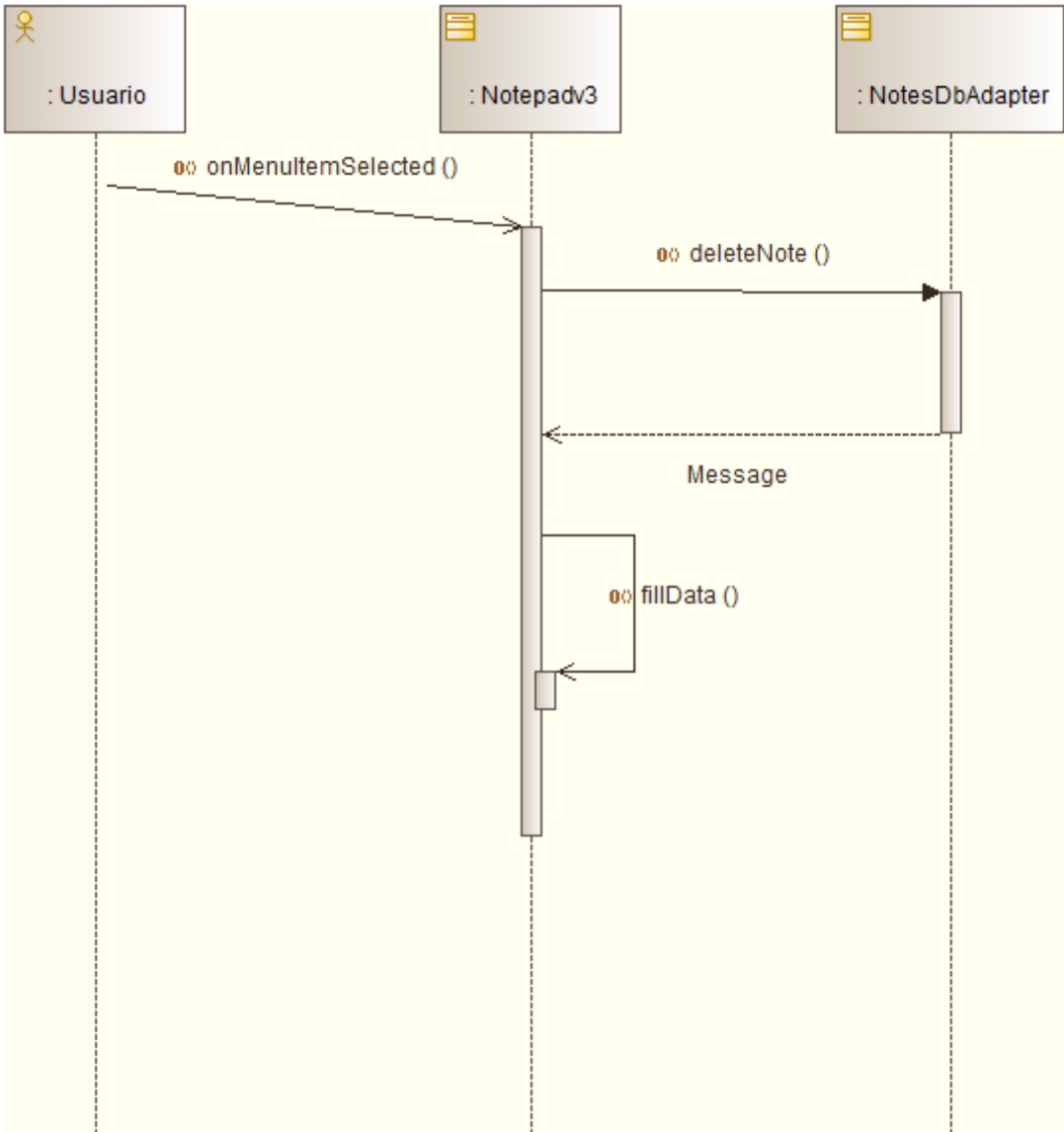
Crear nota



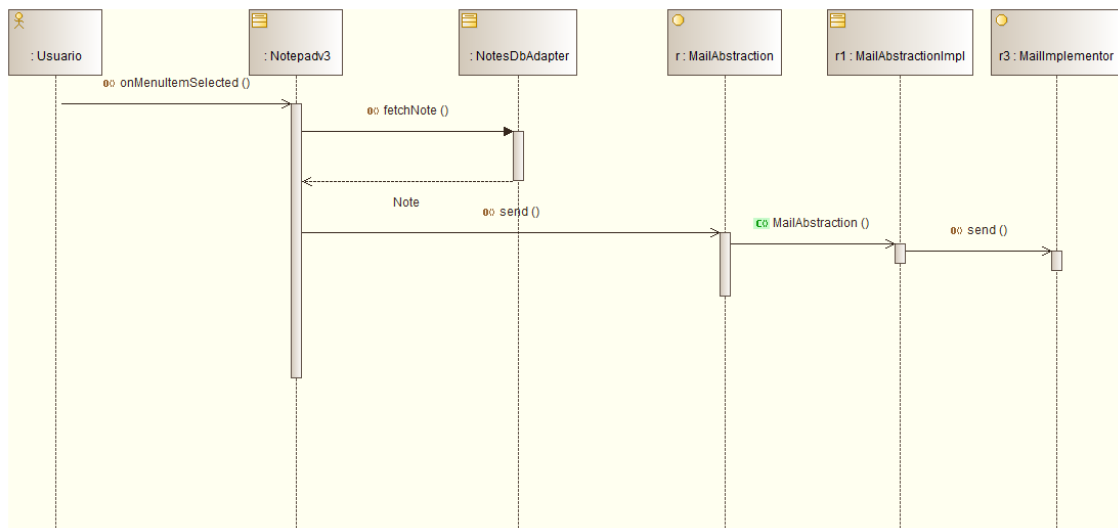
Modificar nota:



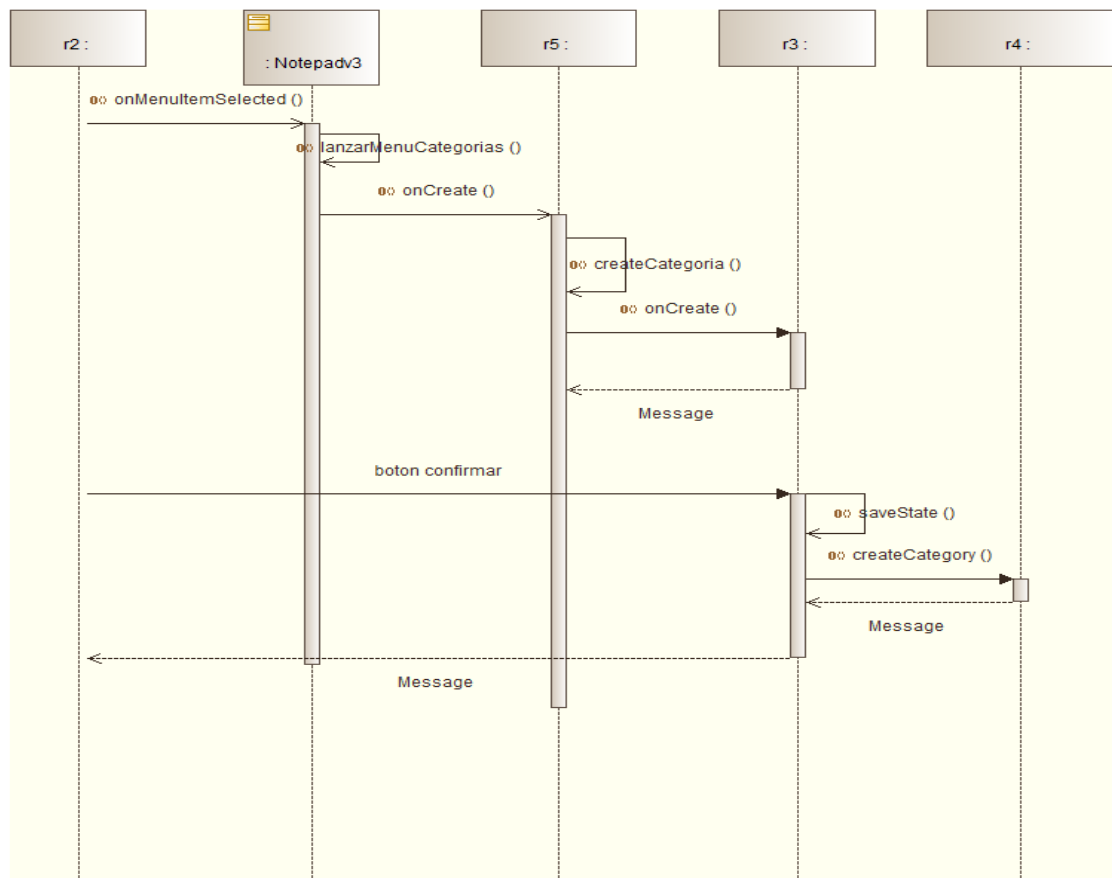
Borrar nota:



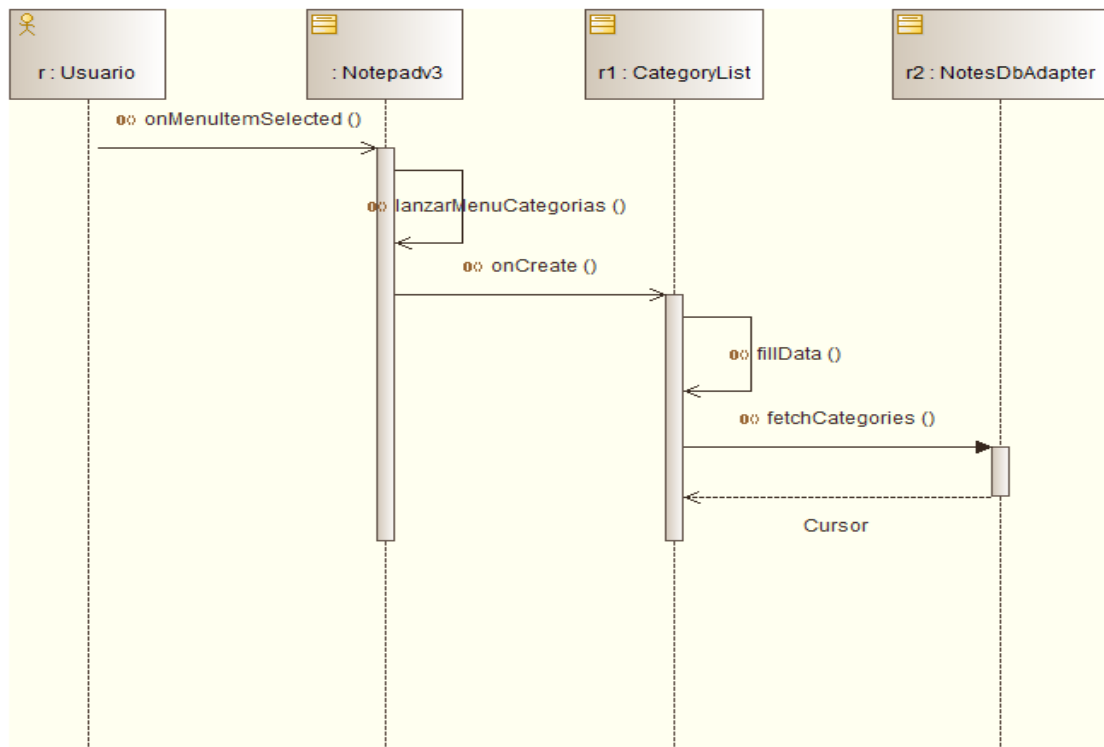
Mandar mail:



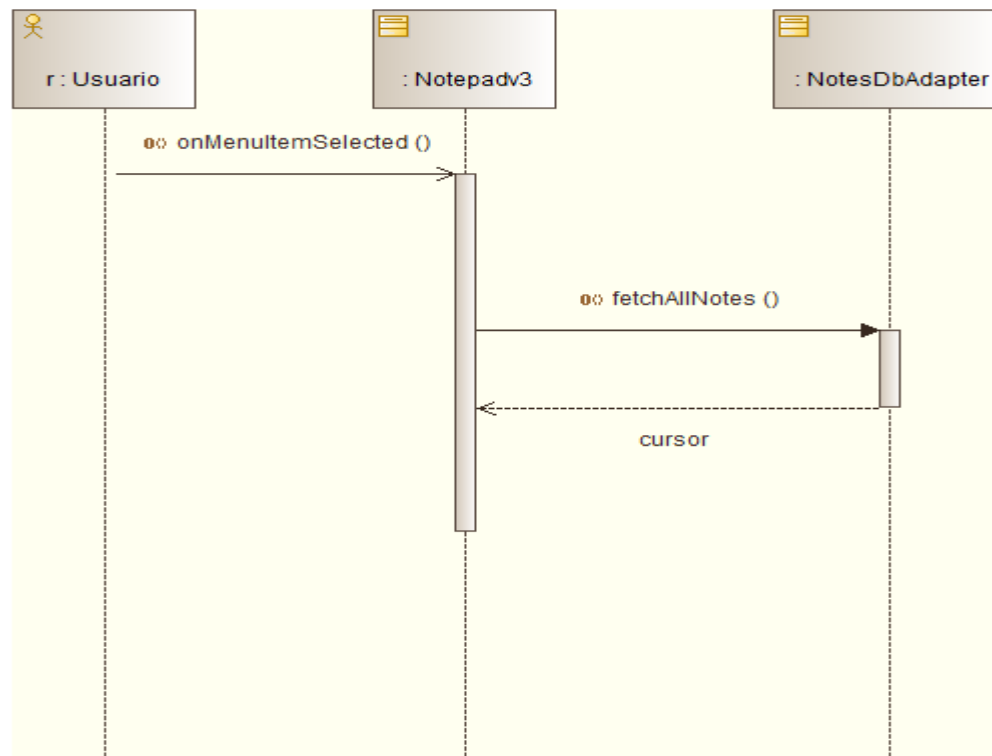
Crear categoría:



Listar categorías:

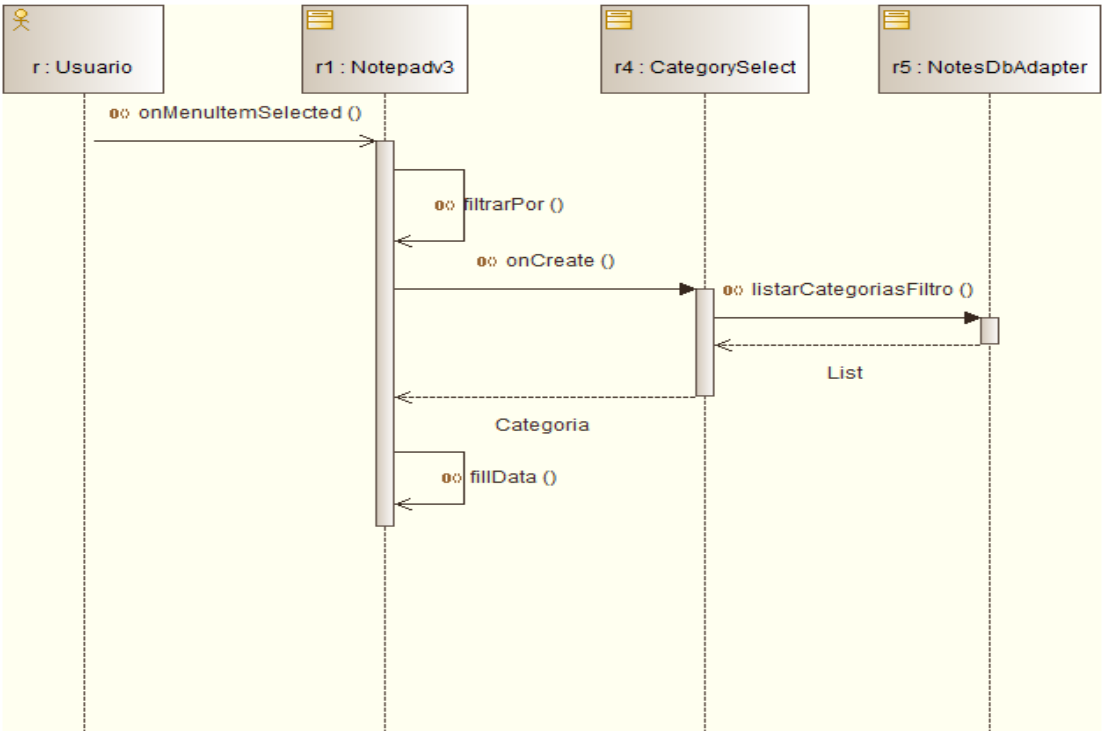


Listar todas las notas:

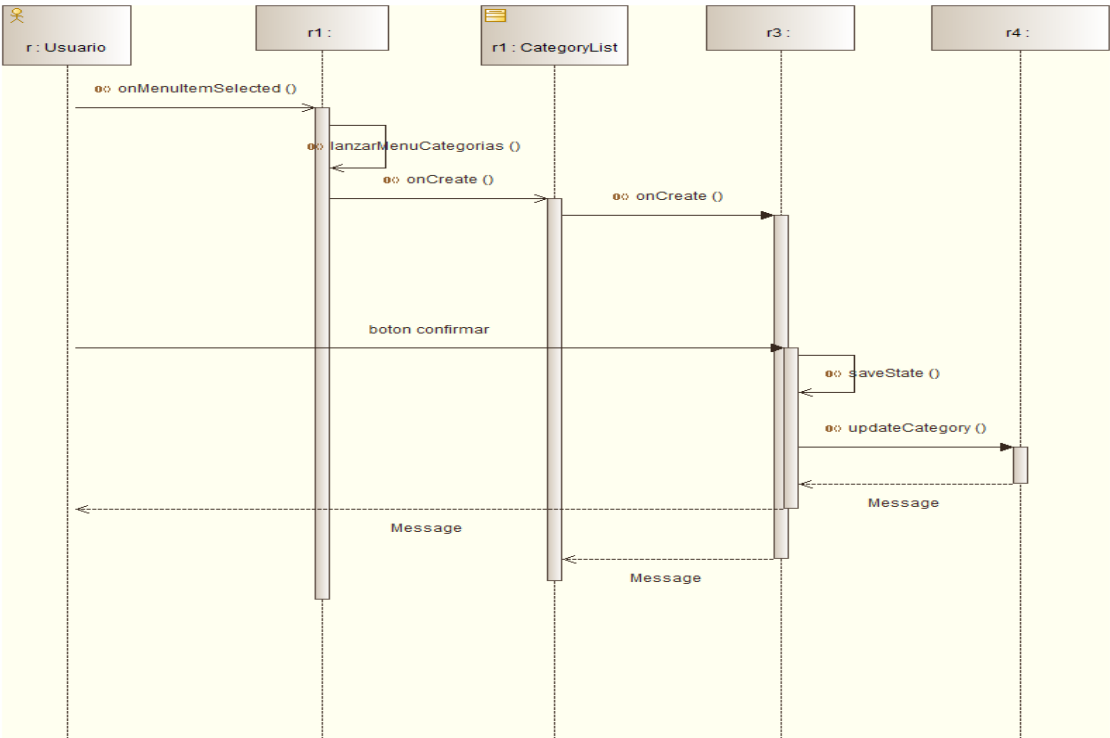


Listar una categoría:

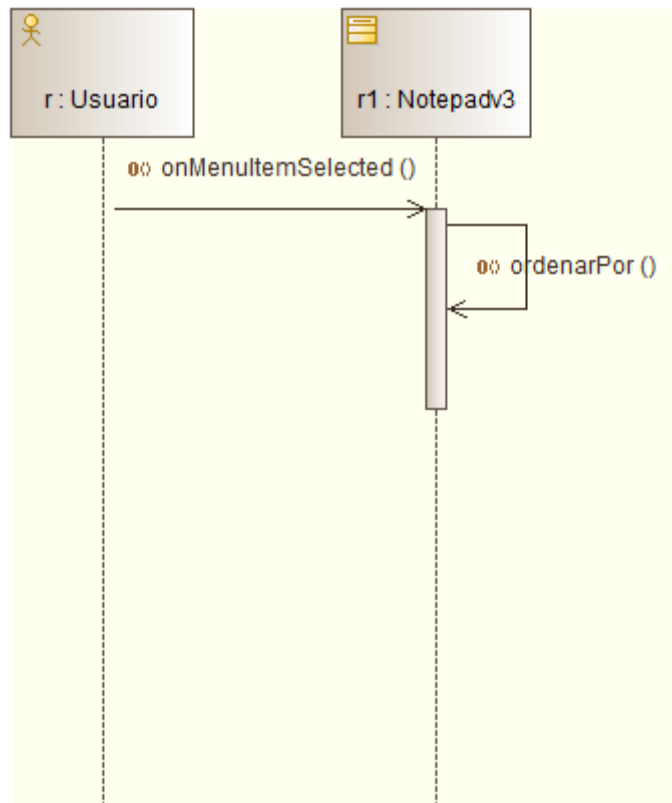
Modific



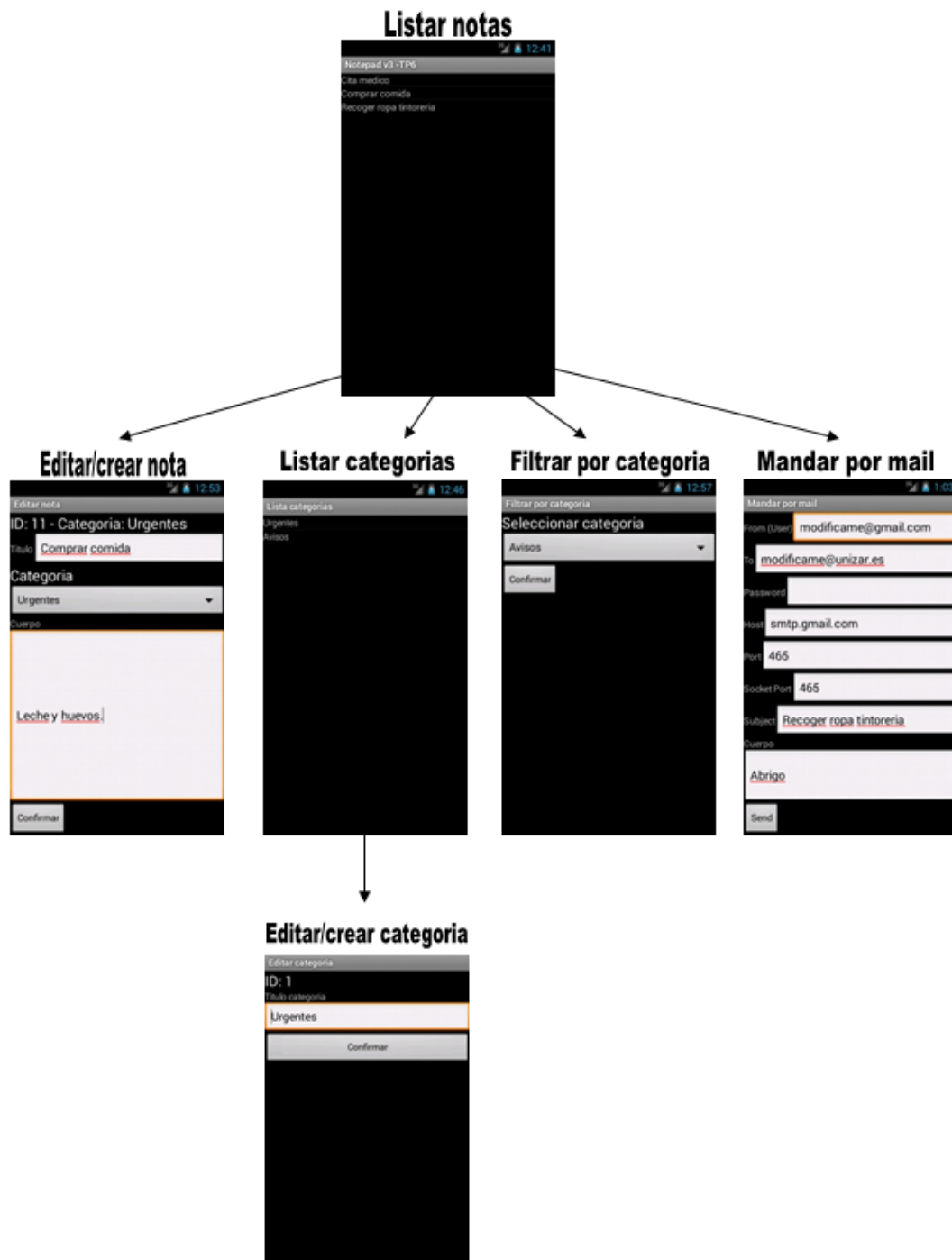
Modificar categoría:



Ordenar por titulo/categoría



MAPA DE NAVEGACIÓN



DISEÑO DEL SISTEMA

DIAGRAMA DE CLASES

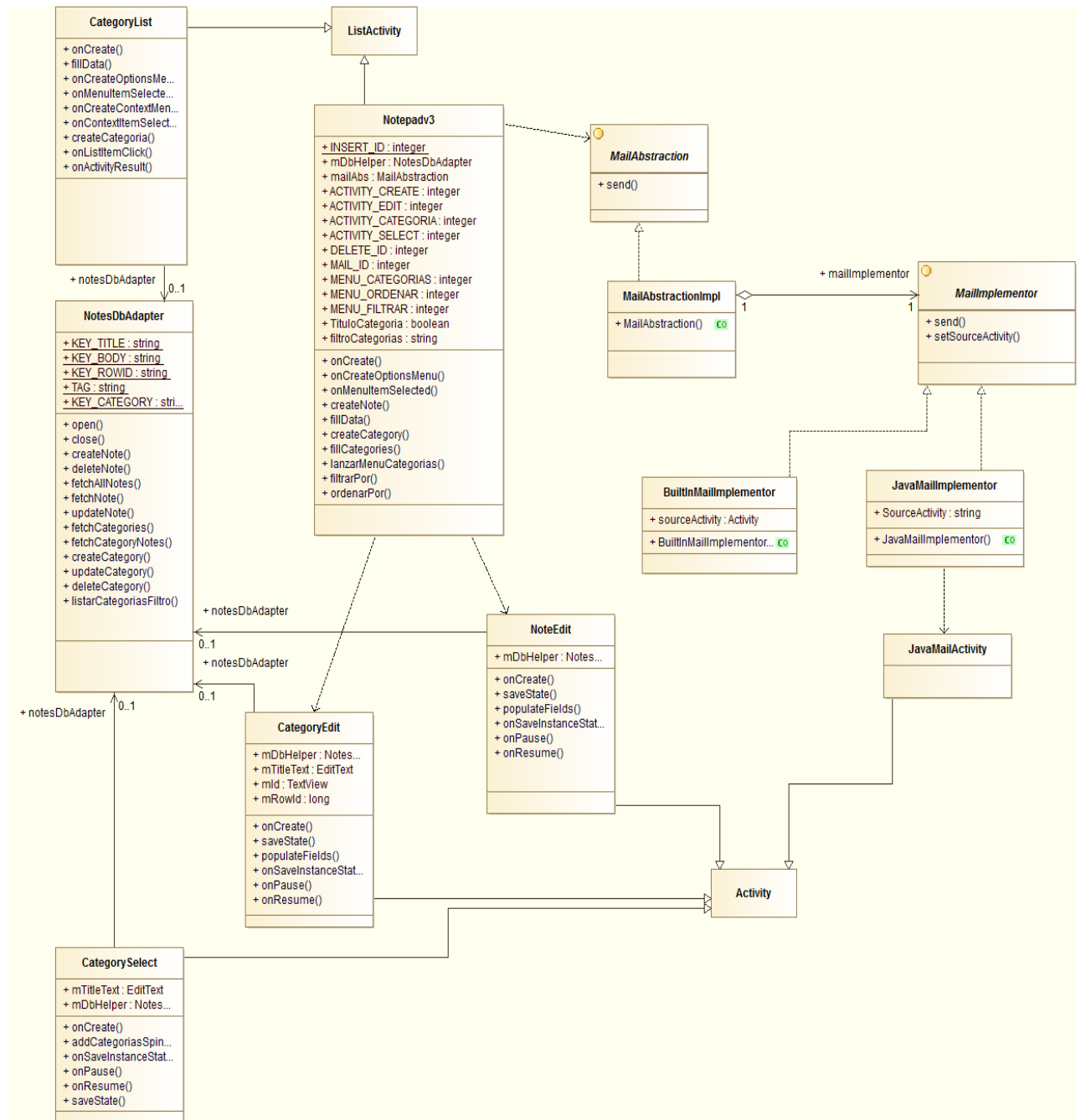


DIAGRAMA DE PAQUETES

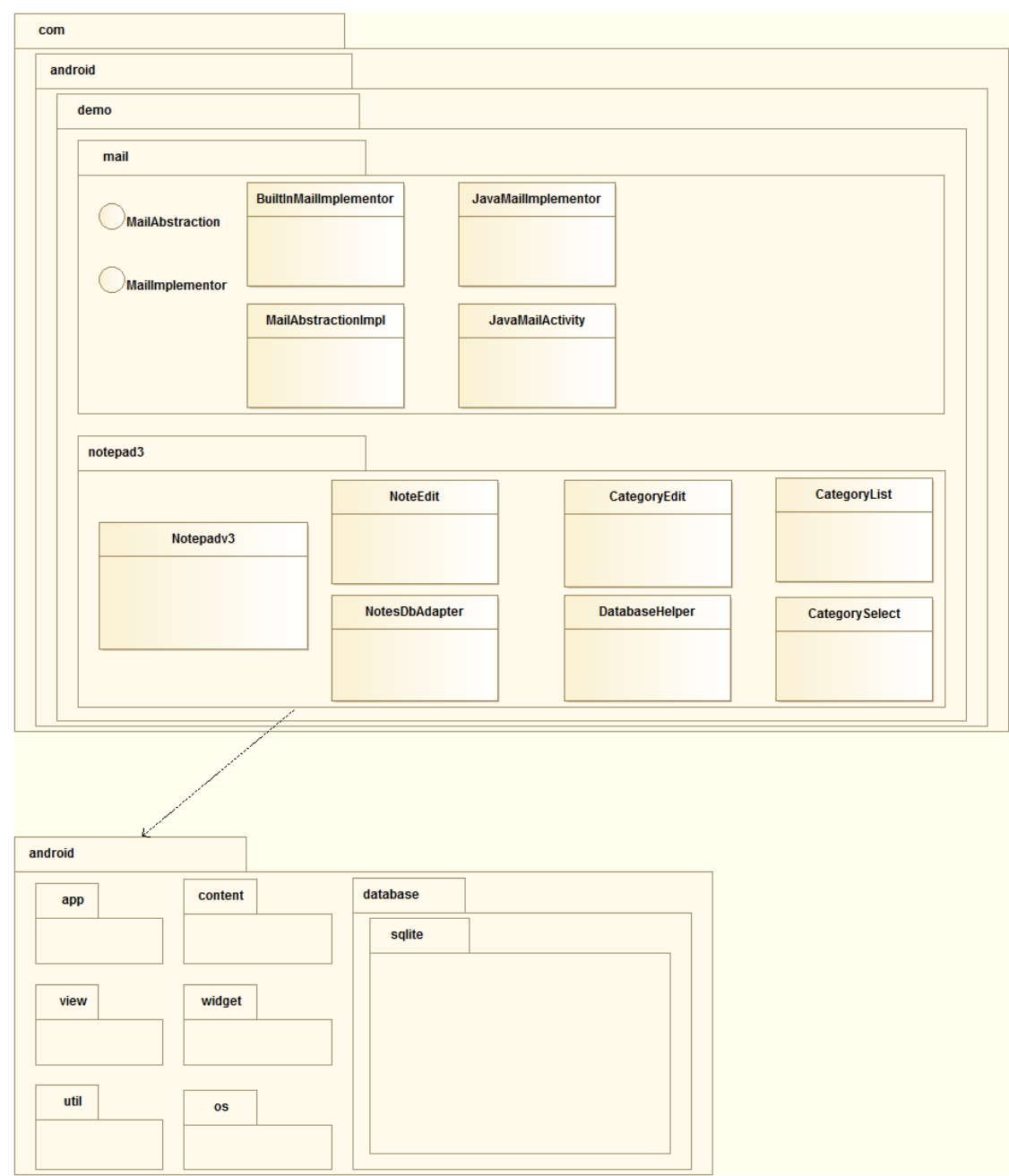


DIAGRAMA DE COMPONENTES

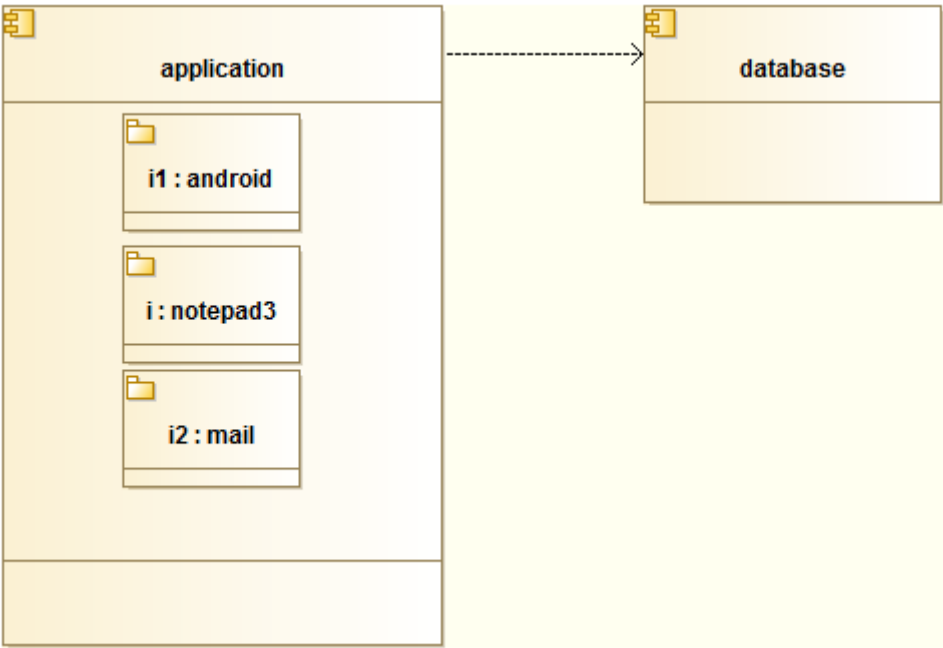
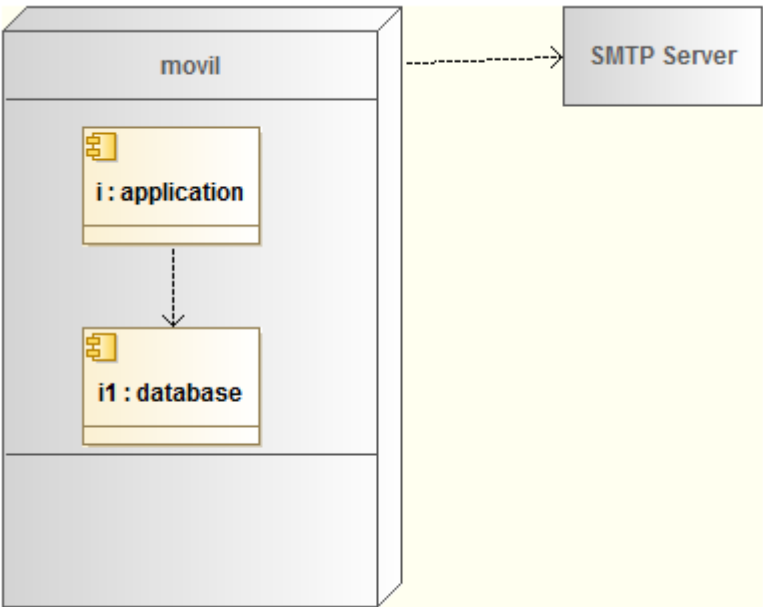


DIAGRAMA DE DESPLIEGUE



DISEÑO DE OBJETOS

Hemos utilizado el patrón bridge para la implementación del sistema de envío por mail.

Hemos utilizado el patrón adapter en la clase NotesDbAdapter.

MODELO RELACIONAL BASE DE DATOS

Hemos optado por disponer de 2 tablas, una contendrá las notas y la otra contendrá las categorías, de tal forma que nos quedarían así:

NOTAS:

| ID NOTA(Clave primaria) | TITULO(No nulo) | CUERPO | NOMBRE CATEGORIA |
|-------------------------|-----------------|------------------|------------------|
| 1 | compra | Huevos, leche... | tareas |
| 2 | | | ... |

CATEGORIAS

| ID CATEGORIA(Clave primaria) | NOMBRE CATEGORIA(No nulo) |
|------------------------------|---------------------------|
| 1 | tareas |

De tal manera que tendremos de clave para relacionar ambas tablas el nombre de la categoría, con esto podremos ordenar las notas por categoría, mostrar solo las notas de una determinada categoría...

PRUEBAS

Pruebas – técnica de particiones de equivalencia

Las particiones de equivalencia para las notas validas son las siguientes:

- Para crear una nota

title != null y title.length() > 0

body != null

- Borrar una nota

rowId > 0

- Actualizar la nota

rowId > 0

title != null y title.length() > 0

body != null

Para comprobar su correcto funcionamiento hemos insertado manualmente notas que parte de los parámetros estaban fuera del rango de equivalencia, los cuales han sido desechados por los diferentes métodos de actualización de la base de datos.

Pruebas – Volumen

Hemos creado la clase ‘Test’ que genera una carga de 1000 notas, cada nota con un Id incremental de 0 a 1000. El resultado que hemos obtenido es que no ha dado ningún error.

Pruebas – Sobrecarga

Para realizar la prueba de sobrecarga hemos creado un método en la clase “Test” con lo cual generamos notas con cuerpo creciente. Cada iteración duplicamos la longitud del cuerpo, comenzando en longitud 10. La ejecución de la prueba nos arroja fallos a partir de notas con longitud superior a 10^{18} .

BIBLIOGRAFIA

Apuntes de la asignatura de IS

Apuntes de la asignatura de Bases de datos

Wikipedia