



**Universidad de Valladolid**

Escuela de Ingeniería Informática

**Trabajo Fin de Grado**

Grado en Ingeniería Informática

(Mención Ingeniería de Software)

**Desarrollo de una aplicación  
Android para gestionar  
información turística de  
localidades**

Autor:

**D. Daniel Alderete Cillero**

Tutor:

**D. Joaquín Adiego Rodríguez**



# Índice

Índice	I
<b>1 Resumen</b>	<b>1</b>
<b>2 Abstract</b>	<b>2</b>
<b>3 Introducción y Objetivos</b>	<b>3</b>
<b>4 Estado del Arte</b>	<b>7</b>
4.1 Android . . . . .	7
4.2 Android Room . . . . .	9
4.3 Google Maps Android API . . . . .	10
4.4 Google Maps Directions API . . . . .	10
4.5 Google Location Manager . . . . .	10
<b>5 Análisis y Diseño</b>	<b>12</b>
5.1 Especificación de la Aplicación . . . . .	12
5.2 Requisitos funcionales . . . . .	15
5.3 Requisitos no funcionales . . . . .	19
5.4 Actores . . . . .	19
5.5 Casos de Uso . . . . .	20
5.6 Modelo de Dominio . . . . .	33
<b>6 Implementación</b>	<b>37</b>
6.1 Prototipo en papel . . . . .	37
6.2 Modelo de Datos . . . . .	39
6.3 Modelo de Dominio . . . . .	40
6.3.1 Dominio . . . . .	40
6.3.2 Persistencia . . . . .	41
6.3.3 Negocio . . . . .	42
6.3.4 Preferencias . . . . .	44
6.3.5 Utilidades . . . . .	45
6.4 Casos de Uso . . . . .	46
6.4.1 Iniciar Aplicación . . . . .	46
6.4.2 Actualizar Base de Datos . . . . .	49
6.4.3 Abrir Mapa . . . . .	50
6.4.4 Agregar Lugar . . . . .	51
6.4.5 Editar Lugar . . . . .	53
6.4.6 Eliminar Lugar . . . . .	54

6.4.7	Trazar Ruta a Lugar . . . . .	55
6.4.8	Mostrar Lugar en Mapa . . . . .	57
6.4.9	Llamar número . . . . .	57
6.4.10	Abrir correo . . . . .	57
6.4.11	Abrir web . . . . .	58
6.4.12	Mostrar Información de Lugar . . . . .	58
6.4.13	Mostrar Rutas . . . . .	59
6.4.14	Ver Ruta . . . . .	60
6.4.15	Agregar Ruta . . . . .	60
6.4.16	Editar Nombre de Ruta . . . . .	62
6.4.17	Eliminar Ruta . . . . .	63
6.4.18	Mostrar Ruta en Mapa . . . . .	64
6.4.19	Editar Ruta . . . . .	65
6.4.20	Compartir Ruta . . . . .	66
6.4.21	Importar Ruta . . . . .	67
6.4.22	Mostrar Lugares cercanos . . . . .	68
6.4.23	Realizar búsqueda de lugares . . . . .	69
6.5	Arquitectura y Despliegue . . . . .	70
6.6	Formato de archivo .intour . . . . .	71

## **7 Pruebas 73**

7.1	Metodología utilizada . . . . .	73
7.2	Resultado de las pruebas . . . . .	74
7.2.1	Iniciar Aplicación . . . . .	74
7.2.2	Actualizar base de datos . . . . .	75
7.2.3	Abrir Mapa . . . . .	75
7.2.4	Agregar Lugar . . . . .	77
7.2.5	Editar Lugar . . . . .	78
7.2.6	Eliminar Lugar . . . . .	79
7.2.7	Trazar Ruta a Lugar . . . . .	79
7.2.8	Mostrar Lugar en Mapa . . . . .	80
7.2.9	Llamar Número . . . . .	81
7.2.10	Abrir Correo . . . . .	81
7.2.11	Abrir web . . . . .	82
7.2.12	Mostrar Información de Lugar . . . . .	82
7.2.13	Mostrar Rutas . . . . .	83
7.2.14	Ver Ruta . . . . .	83
7.2.15	Agregar Ruta . . . . .	83
7.2.16	Editar Nombre de Ruta . . . . .	84
7.2.17	Eliminar Ruta . . . . .	85



7.2.18	Mostrar Ruta en Mapa . . . . .	85
7.2.19	Editar Ruta . . . . .	87
7.2.20	Compartir Ruta . . . . .	88
7.2.21	Importar Ruta . . . . .	89
7.2.22	Mostrar Lugares Cercanos . . . . .	90
7.2.23	Realizar búsqueda de lugares . . . . .	91
<b>8</b>	<b>Conclusiones y Trabajo Futuro</b>	<b>92</b>
<b>9</b>	<b>Referencias</b>	<b>94</b>
<b>A</b>	<b>Distribución de contenidos del soporte digital</b>	<b>96</b>
<b>B</b>	<b>Manual de Instalación</b>	<b>96</b>
<b>C</b>	<b>Manual de Usuario</b>	<b>97</b>
C.1	Descripción . . . . .	97
C.2	Iniciar la aplicación . . . . .	97
C.3	Pantalla inicial . . . . .	98
C.4	Preferencias . . . . .	100
C.4.1	Tipo de mapa . . . . .	100
C.4.2	Visibilidad . . . . .	101
C.4.3	Modo de la Ruta . . . . .	101
C.4.4	Color de la ruta . . . . .	101
C.4.5	Lugares Cercanos . . . . .	102
C.5	Mapa . . . . .	103
C.6	Información de Lugar . . . . .	109
C.7	Lista Lugares . . . . .	111
C.8	Edición de Lugar . . . . .	112
C.9	Lista Rutas . . . . .	114
C.9.1	Mostrar en el mapa . . . . .	116
C.9.2	Compartir ruta . . . . .	116
C.9.3	Cambiar Nombre . . . . .	117
C.9.4	Editar . . . . .	117
C.9.5	Eliminar . . . . .	120
C.10	Edición de nombre de Ruta . . . . .	120
C.11	Acerca de . . . . .	122
<b>D</b>	<b>Diagramas de acceso a datos</b>	<b>123</b>



# 1. Resumen

No cabe duda de que uno de los campos que más crecimiento ha experimentado dentro del sector de las TICs ha sido el de las tecnologías móviles, gracias en parte, al auge de Android desde su aparición. Hoy resulta impensable concebir el mundo sin un dispositivo móvil, ya sea *smartphone* o tableta, apenas a unos centímetros de nuestra mano. Además, ya no son los dispositivos de principios de siglo, si no que se pueden empezar a considerar ordenadores en miniatura por derecho propio, lo que no hace más que aumentar las infinitas posibilidades que ofrecen. Por otra parte, en España, el turismo es una de las principales fuentes de ingresos, y se puede argumentar que es uno de los motores de nuestra economía. Este entorno es extrapolable a Castilla y León, aunque con un enfoque más cultural, ya que no se dispone del atractivo del llamado «Turismo de Sol y Playa». El objetivo del trabajo es unir estas dos situaciones, desarrollando una aplicación para móviles, InfoTourist, utilizando las facilidades de desarrollo proporcionadas por Android, que sirva para promocionar el turismo cultural en Castilla y León; realizando funciones que actualmente no proporcionan las aplicaciones de la Play Store de similares características. No solo se pueden consultar el inmenso listado de bienes culturales de Castilla y León (Monumentos, Museos, Oficinas de Información Turística) y acceder a su información, sino que se permite la creación de rutas y su trazado, pudiéndose compartir con otros usuarios, y mostrar la ubicación del usuario. En esta memoria se documenta todo el proceso de desarrollo de la aplicación, y se incluyen todos los diagramas, tablas y figuras para poder entenderlo, así como toda la información que se ha considerado necesaria.

## 2. Abstract

There is no doubt that, in the field of ICTs, one of the sectors that has experienced bigger development has been that of Mobile technologies, mostly due to Android boom since its apparition. Today, it seems unthinkable to conceive the world without mobile devices, being them *smartphones* or tablets, further than a few centimetres away from our hands. Furthermore, these devices aren't by any way similar to those from the turn of the century, but truly mini-computers on their own right, which increases the infinite possibilities they offer even more. Moreover, in Spain, tourism is one of the principal sources of income at state level, and without doubt one of the pillars of our economy. This environment is applicable to Castilla y Leon, albeit with a more cultural focus, as «Seaside Tourism» appealing is not available. The main objective of this work is to join these two situations, developing a mobile application, InfoTourist, using Android ease of development, which would promote Cultural Tourism in Castilla y León; adding functions that cannot be found currently in similar apps from the Play Store. Not only the immense list of cultural buildings from Castilla y León can be seen (Monuments, Museums, Tourist Offices) and their information be accessed, but routes can be created and drawn, adding the ability to share them with other users, and the user's location can be shown. This report documents the whole development process, and includes every diagram, table and figure needed for its comprehension, as well as all the information which has been considered necessary.

### 3. Introducción y Objetivos

Actualmente, el turismo es una de las mayores fuentes de ingresos de la economía española, generando riqueza para el conjunto de la población y empleando al 13'9 % de la población activa[3]. Estos datos nacionales son extrapolables a nivel autonómico, habiéndose superado los 8 millones de turistas por primera vez en Castilla y León durante el año 2017, lo que supone un incremento del 12'67 % respecto a las cifras de 2016. Del número total, alrededor del 20 % eran turistas provenientes del extranjero, perteneciendo el resto al territorio nacional. Además, nuestra comunidad continúa siendo líder en el ámbito del turismo rural. El impacto económico ha sido de 2000 millones de euros, con un incremento del 3'2 % en comparación con el año 2016. Los turistas prefieren los meses de julio y agosto para realizar sus visitas. El turismo que se realiza es en su mayor parte de carácter cultural, con mención especial al turismo gastronómico[9].

Realizando búsquedas en la Play Store de Google de aplicaciones relacionadas con el turismo en Castilla y León se comprobó que no existían aplicaciones que permitieran mostrar los puntos de interés de nuestra comunidad en conjunto, limitándose en los casos que se encontró a localidades específicas: por ejemplo, mostrando sólo los monumentos de Valladolid. Además, las aplicaciones que se encontraron mostraban sobre todo hoteles, restaurantes, aparcamientos de coches, obviando los lugares de interés propios de las ciudades. Otro formato de aplicación encontrado fue el de «Agenda Cultural» con información sobre los eventos, festivales, etc.

Analizando esta situación, se llega a las siguientes conclusiones:

- Dado que ya existen diversas aplicaciones que muestran información de restaurantes, hoteles y demás sitios relacionados, se descarta esta opción para realizar la aplicación, por estar ya elaborada.
- Igualmente, en el sector de las llamadas «Agendas Culturales» se encuentran abundantes ejemplos de aplicaciones que cumplen esta función de manera satisfactoria, por lo que también se descarta.
- De las aplicaciones relacionadas con los lugares de interés, la inmensa mayoría están circunscritas a localidades específicas, y no llegan a mostrar todos los monumentos o museos de la localidad, limitándose a los más importantes.
- Varias aplicaciones ofrecen la posibilidad de seguir rutas ya establecidas por dichos lugares, sin posibilidad de agregar rutas propias del usuario.
- El turismo cultural es importante para Castilla y León, y sin embargo no existen aplicaciones de ámbito comunitario para mostrar los lugares de interés de la comunidad.
- No existe una aplicación unificada que muestre oficinas de información turística, limitándose como en casos anteriores a mostrar las de la localidad o provincia específica.

- De las aplicaciones encontradas, muy pocas mostraban la ubicación actual del usuario, y tampoco permitían trazar el itinerario a seguir desde esta ubicación a cualquiera de los lugares mostrados.

Por lo tanto, se establecen los siguientes objetivos que deberá cumplir la aplicación:

- Mostrar en un mapa todos los monumentos, museos y oficinas de información turística de Castilla y León.
- Mostrar en el mapa la ubicación actual del usuario (si éste lo permite).
- Mostrar cierta información de cada uno de los lugares, siempre que sea posible.
- Permitir realizar ciertas acciones sobre el lugar, siempre que se encuentre la información necesaria (por ejemplo, llamar al teléfono de contacto o abrir la página web).
- Permitir al usuario encontrar la ruta que le lleve desde su posición hasta un lugar determinado, tanto a pie como en coche.
- Permitir la búsqueda de lugares, usando un buscador.
- Permitir al usuario definir y mostrar los lugares que se encuentren cercanos a su posición.
- Permitir al usuario crear rutas con los lugares que él especifique, permitiendo más tarde editarlas, compartirlas e importarlas. Una vez creada se puede trazar sobre el mapa.
- Permitir al usuario definir sus propios lugares personalizados, pudiendo incorporarse a rutas.
- Incorporar traducciones a inglés, francés y alemán de la aplicación, al ser los turistas internacionales un porcentaje importante del total.
- Realizar la aplicación en Android debido a su carácter gratuito y a las facilidades de programación que ofrece.

Se decide entonces, partir de una aplicación con características similares que se realizó durante el curso 2016/2017 para la asignatura de sistemas móviles, incorporando cuantiosas mejoras, corrigiendo errores, solucionando problemas e incorporando nuevas funcionalidades.

Se establece un plan de gestión que, a grandes rasgos es el siguiente:

- Semanas 1 y 2 de febrero: refresco de conocimientos de Android y aprendizaje de nuevas opciones (por ejemplo, uso de Android Room para bases de datos).
- Semana 3 de febrero: análisis y diseño de la aplicación.
- Semanas restantes de febrero, y semanas 1 a 4 de marzo: desarrollo e implementación de la aplicación.

- Semana 5 de marzo: pruebas sobre la aplicación completa y corrección de errores (bugfixing).
- Abril y mayo: escritura de la memoria.

Las primeras semanas se dedicaron a recordar conceptos fundamentales de Android y a familiarizarse de nuevo con sus características y con las nuevas opciones que permite tanto Android como Android Studio, al igual que al estudio de nuevas opciones que se han incorporado a lo largo de 2017. Posteriormente se analizó la aplicación y se diseñó.

Se decide seguir un ciclo de desarrollo-implementación-pruebas en el que se prueba la funcionalidad añadida durante un periodo no mayor de 2 días, con pruebas de la aplicación entera realizadas los domingos de cada semana de desarrollo. Se implementa primero la funcionalidad propiamente dicha, sin dedicar mucho tiempo a la interfaz. Una vez completado, se modifica la interfaz para hacerla usable para el usuario. Por último se traduce la aplicación. A mayores se dedicó la última semana de Marzo a la realización de pruebas más exhaustivas sobre la aplicación ya terminada y se corrigieron todos los errores que se hallaron. Durante estas pruebas se contó con la participación de usuarios reales, que testearon la aplicación. Por último se decide escribir la memoria.

En un principio se utilizó Git como sistema de control de versiones y GitHub para el almacenamiento del proyecto, pero tras una serie de errores de autenticación y limitación en la cantidad de archivos para subir, se optó por un proyecto en BitBucket, que hasta este momento se ha comportado de forma completamente satisfactoria. El enlace del proyecto es [https://bitbucket.org/R\\_Daniel3/infotourist-tfg/overview](https://bitbucket.org/R_Daniel3/infotourist-tfg/overview)

Se contactó con la Junta de Castilla y León, ya que durante el desarrollo se descubrieron dos errores en uno de los ficheros (en concreto, dos lugares ubicados en Salamanca y Burgos aparecían en Colombia y Estados Unidos, respectivamente). A los pocos días se recibió la respuesta, y se confirmó que, efectivamente, los lugares aparecían en sus coordenadas correctas.

A mediados de marzo se tuvo una reunión con el tutor, en la que se resolvieron varias dudas respecto al diseño de la interfaz, y se realizaron varias propuestas de mejora para la aplicación (cambiar el color en el modo de edición) que se incorporaron posteriormente.

A lo largo de esta memoria se comenzará detallando el conocido como *Estado del Arte*, la situación actual de la tecnología utilizada. Se continúa describiendo la fase de análisis de la aplicación, en la que se describe toda su funcionalidad, requisitos, casos de uso y modelo de dominio. En la sección de implementación se explica como se tradujo lo anterior a Android, explicando la funcionalidad en detalle, e incluyendo el prototipo en papel de la aplicación. La sección de pruebas detalla los tests que se realizaron sobre la aplicación final, con todos los resultados y condiciones de validación. Por último se incluyen las conclusiones extraídas del trabajo y se detalla el trabajo

futuro que se podría realizar sobre la aplicación. En la sección de referencias se refleja toda la bibliografía utilizada, y los respectivos anexos detallan la distribución de contenidos de la memoria USB, y los manuales de instalación y de usuario.



## 4. Estado del Arte

En esta sección se detalla el estado actual de la tecnología usada durante la realización de la aplicación, a saber, Android y las diferentes APIs que se han usado a lo largo del desarrollo. El estado actual de las aplicaciones para móvil similares a ésta queda explicado en la sección anterior.

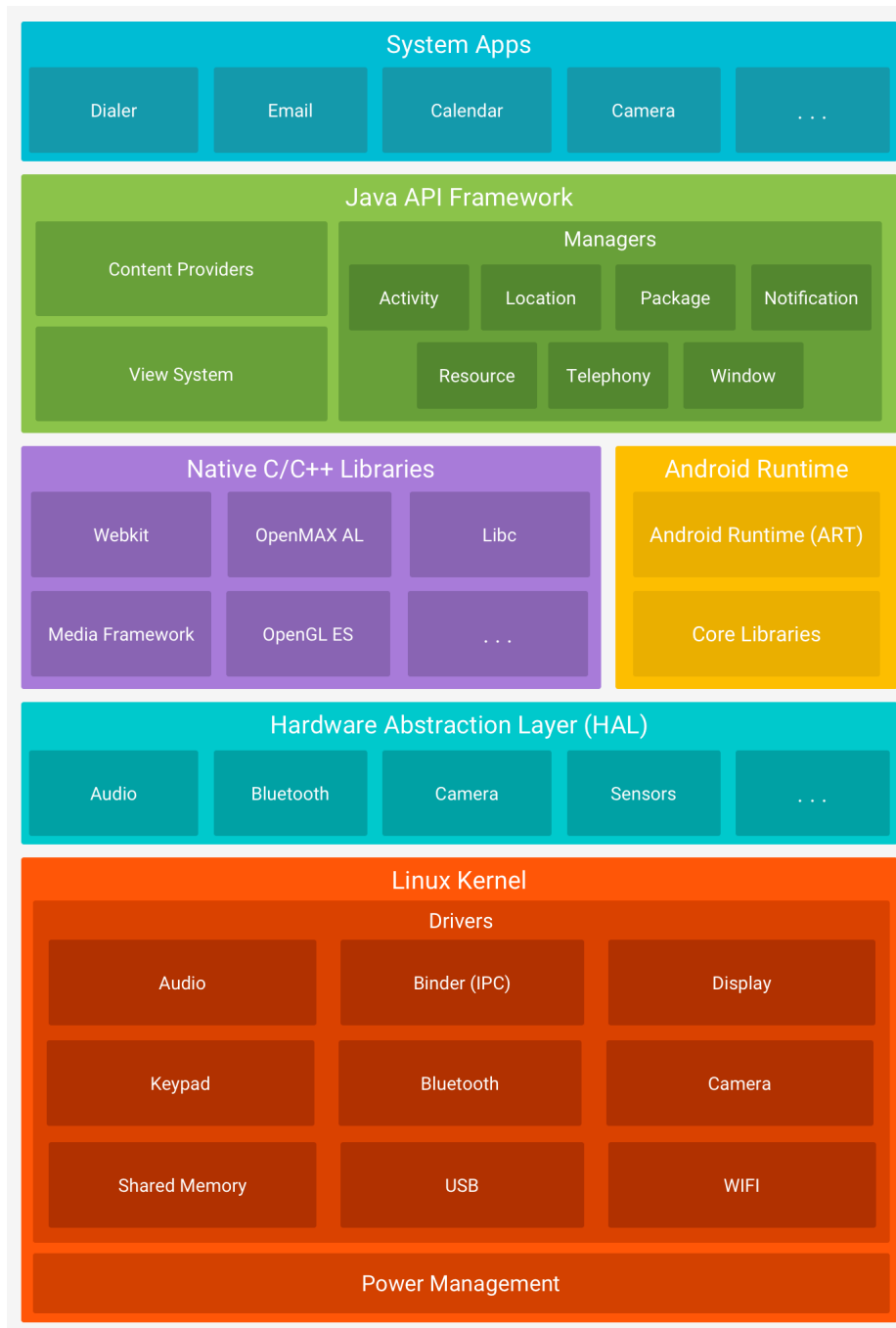
### 4.1. Android

Actualmente, Android es el sistema operativo para dispositivos móviles más usado del mercado, con un 74'24 % de cuota de mercado en Marzo de 2018, cifra claramente superior a su más cercano competidor, iOS, que alcanza un 20'83 %[25]. Las razones por las que este sistema operativo es tan popular pueden encontrarse en sus características:

- Plataforma realmente abierta basada en software libre.
- Adaptable a cualquier tipo de hardware.
- Portabilidad asegurada.
- Arquitectura basada en componentes inspirados en Internet.
- Filosofía de dispositivo conectado siempre a Internet.
- Gran cantidad de servicios incorporados.
- Aceptable nivel de seguridad.
- Optimizado para baja potencia y poca memoria.
- Alta calidad de gráficos y sonido.

Por lo tanto, Android ofrece de forma sencilla y novedosa la posibilidad de implementar aplicaciones para diferentes tipos de dispositivos sin perder potencia por ello[4].

La arquitectura de Android se basa en cuatro capas claramente diferenciadas, como se ve en la figura.



**Figura 1:** Arquitectura de Android

- Núcleo Linux: formado por el sistema operativo Linux 2.6. Proporciona servicios esenciales como la seguridad, manejo de memoria... Esta capa actúa como abstracción entre el hardware y el resto de la pila, por lo que depende del hardware.
- HAL: *Hardware Abstraction Layer*, brinda interfaces estándares que exponen las capacidades de hardware del dispositivo al framework de la API de Java. HAL consiste en varios módulos de biblioteca y cada uno de estos implementa una interfaz para un tipo específico de componente de hardware.
- Runtime: basado en las máquinas virtuales de Java. Dadas las limitaciones de los dispositivos móviles, se utiliza Dalvik, una JVM adaptada a estas características. Cada aplicación corre

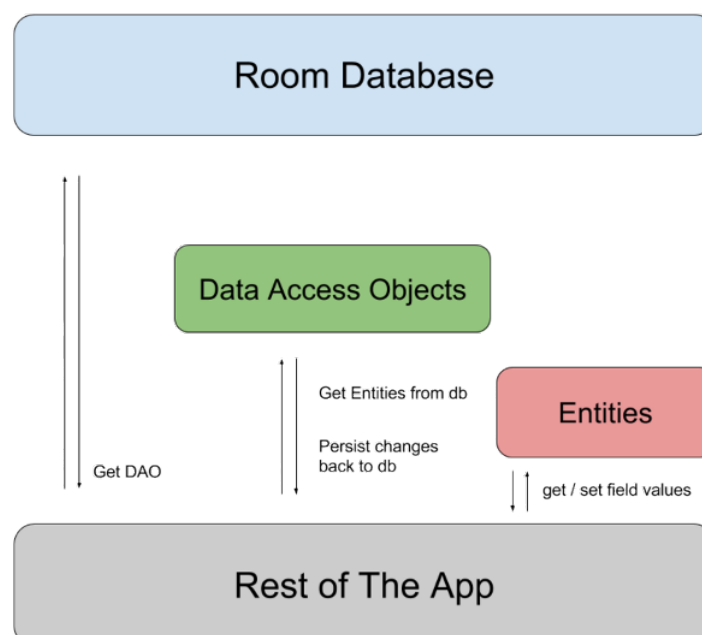
su propio proceso Linux en su instancia Dalvik. A partir de la API 21 se cambia a ART, que aumenta la eficiencia. Core libraries incluye la mayoría de librerías disponibles para Java.

- Librerías Nativas: librerías C/C++ usadas en los componentes, compiladas en el código nativo del procesador.
- Java API Framework: conjunto de aplicaciones instaladas en Android y las herramientas necesarias para poder ejecutarlas.

## 4.2. Android Room

Android Room proporciona una capa de abstracción por encima de SQLite (el S.G.B.D. de Android) para permitir un acceso fluido a la base de datos sin disminuir las capacidades de SQLite[16]. Google recomienda el uso de Room en vez de trabajar directamente con la API para Android de SQLite. Room tiene tres elementos principales:

- Database: representa a la base de datos y sirve como punto de acceso a esta. Se utiliza la anotación @Database, debe ser una clase abstracta que extiende de RoomDatabase, incluye las entidades asociadas a la base de datos, contiene un método que devuelve el DAO por cada Entity, debe estar anotado con @Dao. Se recomienda utilizar el patrón singleton.
- Representa una tabla de la base de datos.
- DAO: contiene los métodos de acceso a datos.



**Figura 2:** Arquitectura de Room

### 4.3. Google Maps Android API

Esta API proporciona a la aplicación la posibilidad de mostrar un mapa, ya sea en modo normal, vista satélite, con edificios en tres dimensiones, etc[22]. Además ofrece grandes opciones de personalización en la información mostrada y en el diseño de iconos, acciones y otros elementos que conforman el mapa. Para poder usarse se necesita una clave gratuita que proporciona acceso a todas las operaciones, si no se dispone aparecerá en blanco el mapa.

### 4.4. Google Maps Directions API

Calcula indicaciones entre ubicaciones, pudiendo especificar el medio de transporte[23], se incluyen transporte público o particular, desplazamiento a pie y en bicicleta. La solicitud del cálculo de una ruta se hace a través de una interfaz HTTP, con las coordenadas de las ubicaciones (o sus nombres) en la URL. Un ejemplo proporcionado por Google es (este ejemplo usa la API de pago, la versión gratuita es igual pero se elimina el parámetro «key»):

- Protocolo: https
- Host: maps.googleapis.com
- Dirección: /maps/api/directions/json
- Parámetros:
  - origin: ubicación inicial
  - destination: ubicación final
  - key: clave de API

La respuesta se devuelve en formato Json, que después se puede usar para dibujar la ruta sobre el mapa.

Esta API también permite definir *waypoints* o puntos de paso (hasta 23 en la versión no premium), es decir, puntos adicionales por donde debe pasar la ruta entre el destino inicial y el final.

### 4.5. Google Location Manager

Proporciona el servicio de localización de la ubicación del dispositivo del usuario. Permite a las aplicaciones que lo usan obtener actualizaciones periódicas acerca de la ubicación geográfica

del dispositivo. También permite lanzar un *Intent* definido por la aplicación cuando el usuario se encuentra en las proximidades de una localización geográfica determinada. Necesita de los permisos de Android para acceder a la localización, y dependiendo de ellos variará su precisión a la hora de mostrar la ubicación[24].

## 5. Análisis y Diseño

### 5.1. Especificación de la Aplicación

Partiendo de la situación explicada en la sección de Introducción se realiza la especificación de la aplicación.

InfoTourist es una aplicación para móviles con sistema operativo Android que permite ver a sus usuarios todos los Monumentos, Museos y Oficinas de Información Turística que se encuentran en Castilla y León. La información de estos lugares (latitud, longitud, nombre, descripción, horarios y tarifas, dirección, localidad, provincia, correo, tipo, web y teléfono) se obtiene de los ficheros del Portal de Datos Abiertos de la Junta de Castilla y León, incorporándose a la base de datos del dispositivo, implementada en SQLite con Android Room. Se decide que resulta mejor almacenar la información en los dispositivos de los usuarios y recuperarla de local, que almacenarla en un servidor central y que los usuarios la recuperen; al considerarse que el uso sin Internet, a pesar de no contar con todas las funcionalidades disponibles, es preferible a la conexión permanente con dicho servidor. De esta forma, sólo es necesario realizar una única conexión (la primera vez que se abre la aplicación), en la que se descarga la información de los ficheros, permitiendo al usuario actualizar su base de datos cuando desee.

Una vez incorporada la información de los ficheros a la base de datos, se abre el mapa, que muestra los lugares que se recuperen de la BD. Cada lugar puede pertenecer a un tipo determinado (Monumento, Museo, Oficina, Personalizado) y se diferencian por el color de su marcador en el mapa (rojo, azul, amarillo y verde respectivamente). Los lugares mostrados se agrupan en clusters por proximidad y altura de cámara, es decir, varios lugares que se encuentren muy cercanos aparecerán como un cluster, teniendo que acercarse la cámara para poder ver los lugares individuales. Cada cluster muestra el número de lugares que contiene. Se decide utilizar clusters para no sobrecargar la pantalla del dispositivo con marcadores de lugares, cuando estos se encuentren visualmente muy próximos entre sí, y para que al alejar la cámara no se repita esa situación.

Al abrirse el mapa, si se encuentra la localización del usuario, la cámara se desplaza hasta sus coordenadas, si no se desplaza hasta una posición determinada. Se usan los servicios de Google Maps para mostrar el mapa y el proveedor de GPS de Google para obtener la localización, siempre que se proporcione el permiso. Se permite mover la cámara a la posición del usuario en cualquier momento. El mapa permite alejar y acercarse la cámara dentro de un rango establecido por Google.

Al pulsar sobre un marcador se muestra una ventana informativa donde aparece el nombre del lugar y su tipo. Se habilita la opción de trazar ruta hasta ese lugar, usando la API de Google para obtenerla y permitiendo especificar el modo (a pie o en coche); si no se encuentra la ruta se informa de ello al usuario. Pulsando sobre la ventana de información se accede a la visión detallada de la

información de ese lugar. Si el lugar posee número de teléfono, página web o dirección de correo electrónico se permite al usuario llamar (si se tiene permiso), abrir la web o escribir un correo respectivamente. Además se puede mostrar el lugar en el mapa (la cámara se desplaza hasta sus coordenadas y se muestra su ventana de información) y trazar la ruta a ese lugar (esta acción es idéntica a la especificada anteriormente, sólo cambia el lugar desde el que se accede a la opción).

Se permite al usuario agregar lugares propios, de los que se guarda una cantidad de información parecida al resto de lugares (menos los horarios y tarifas), permitiendo al usuario establecer la siguiente información: nombre, teléfono, web, correo, descripción. Se intenta obtener la dirección del lugar por medio de los servicios de geolocalización proporcionados por Google. Una vez agregado, se trata al lugar como cualquier otro. El usuario puede, además, editar y eliminar el lugar en cualquier momento, aunque al eliminar, si el lugar se encuentra agregado a una ruta, se informa y no se permite.

El usuario puede obtener los lugares que se encuentren a una distancia igual o menor a la distancia de búsqueda especificada por él.

El usuario puede realizar búsqueda de valores de texto en los lugares, mostrando los resultados que se hayan obtenido. Si se introduce la misma palabra varias veces, sólo se cuenta una aparición. La aplicación buscará coincidencias en el nombre, la localidad y la descripción del lugar.

La aplicación permite al usuario definir cuantas rutas desee, de las que se almacena su nombre y su itinerario (colección de lugares). Se permite al usuario mostrar todas las rutas almacenadas en la aplicación, si no se encuentra ninguna se informa. El usuario puede agregar rutas, especificando su nombre, y no se permite que dos rutas tengan el mismo nombre, aunque sí el mismo itinerario. Para cada ruta se habilitan una serie de opciones:

- **Mostrar en el mapa:** Se intenta trazar la ruta entre los lugares que componen su itinerario. Si no se encuentra la ruta entre alguno de ellos se informa. Si está activada la localización del usuario se intenta buscar la ruta entre sus coordenadas y el primer lugar de la ruta. Las rutas entre los lugares del itinerario de la ruta se buscan por orden de aparición, no por recorrido óptimo.
- **Compartir Ruta:** se genera un fichero que contiene la ruta y se permite elegir al usuario el medio de compartición: correo electrónico, guardar en dispositivo, etc. (NOTA: actualmente existe un bug al elegir la opción «Guardar en Drive», pero por falta de tiempo no se ha podido investigar a fondo)
- **Cambiar nombre:** se modifica el nombre de la ruta. Si ya existe una ruta con el mismo nombre, se informa y no se permite la edición.
- **Editar:** permite al usuario agregar los lugares que desee a la ruta. Una vez seleccionada se

activa el «Modo de edición», en el cual se permite seleccionar los lugares a agregar, pero no el trazado de rutas ni el visionado de las rutas que se encuentran en la aplicación (todas las demás opciones sí están disponibles). Se modifica el color de la aplicación para que el usuario sepa que está en un modo especial de la aplicación. Si un lugar está agregado a la ruta y se intenta agregar otra vez, no se notifica. El usuario puede confirmar o descartar los cambios en cualquier momento, requiriendo confirmación en este último caso. En caso de existir cambios se incorporan a la base de datos. Se permiten modificar los valores de las preferencias en cualquier momento sin desactivarse el modo de edición por ello.

- Eliminar: se elimina la ruta de la base de datos, requiere confirmación por parte del usuario.

A mayores se pueden importar rutas. En este caso se abre un selector de ficheros donde el usuario selecciona el archivo que contiene la ruta a importar y se intenta importar. El archivo debe cumplir una serie de condiciones para poder ser importado: será de tipo «.intour» y el formato del nombre será NOMBRE DE LA RUTA - SEPARADOR - IDENTIFICADOR DE RUTA. La especificación del formato interno del fichero se detallará más adelante. Si la importación finaliza con éxito, se incorpora la ruta a la base de datos. Si la ruta contiene lugares personalizados que no se encuentran en la base de datos, se agregan.

El usuario puede definir el valor de los valores de configuración que siguen

- Tipo de Mapa: normal, satélite, terreno, híbrido
- Visibilidad de Monumentos, Museos, Oficinas y Personalizado
- Modo de Ruta: a pie, en coche
- Color de Ruta: rojo, verde, azul, amarillo, blanco, negro
- Distancia lugares cercanos: 500m, 1km, 2km, 3km, 5km

Estos valores pueden ser modificados en cualquier momento del uso de la aplicación, realizando las actualizaciones necesarias. Para actualizar los valores relacionados con las rutas será necesario volver a calcularlas.

El idioma por defecto de la aplicación es inglés de Estados Unidos (idioma por defecto de cualquier aplicación Android), y proporciona traducciones al inglés de Reino Unido, alemán, francés y español.

Se decide que la versión mínima de la API de Android sea el nivel 19, y que no haya máxima API, aprovechándose de las funciones de retrocompatibilidad de esta plataforma. Esta decisión se ve motivada al intentar equilibrar las prestaciones (a mayor nivel de API más características) con la cantidad de usuarios que poseen dispositivos con ese nivel. Con esta selección de API se



considera que se alcanza el equilibrio, ya que el nivel 19 incorpora bastante funcionalidad con respecto a las anteriores y se ofrece una amplia cobertura de los dispositivos, dejando apenas un 5% de dispositivos sin posibilidad de uso de la aplicación[19].

A partir de este análisis de la aplicación se detallan los siguientes requisitos, tanto funcionales como no funcionales.

## 5.2. Requisitos funcionales

ID	Nombre	Descripción	Prioridad
RF01	Almacenamiento de datos	El sistema almacenará los datos de los lugares del portal de datos abiertos de la JCYL para los siguientes archivos: relación de monumentos, relación de oficinas de turismo y relación de museos en Castilla y León.	Alta
RF02	Actualización de datos	El sistema permitirá al usuario actualizar los lugares que se encuentren en la base de datos.	Alta
RF03	Tipos de lugares	El sistema deberá permitir distinguir entre cuatro tipos de lugares: Monumentos, Museos, Oficinas de Turismo y Personalizados.	Alta
RF04	Distinción visual del tipo de lugar	El sistema permitirá distinguir visualmente los lugares mostrados en el mapa.	Alta
RF05	Visionado de lugares	El sistema mostrará en un mapa la localización de los lugares.	Alta
RF06	Valores no disponibles	El sistema mostrará «No disponible» en los campos de un lugar que no contengan información.	Media
RF07	Localización del usuario	El sistema podrá usar la ubicación del usuario siempre que este se lo permita, pudiendo operar sin excepciones si esta no es concedida.	Alta
RF08	Creación de lugares	El sistema deberá permitir a los usuarios crear lugares del tipo «Personalizado» en el lugar del mapa donde deseen.	Media

ID	Nombre	Descripción	Prioridad
RF09	Edición de lugares	El sistema permitirá a los usuarios editar lugares únicamente de tipo «Personalizado».	Media
RF10	Borrado de lugares	El sistema permitirá a los usuarios eliminar lugares únicamente del tipo «Personalizado», siempre que no se encuentre agregado a una ruta.	Media
RF11	Información de los lugares	El sistema almacenará los siguientes datos para cada lugar: latitud, longitud, nombre, descripción, horarios y tarifas, dirección, localidad, provincia, correo, tipo, web y teléfono.	Alta
RF12	Visualización de información de lugar	El sistema permitirá mostrar la información de los lugares.	Alta
RF13	Búsqueda de coincidencias	El sistema permitirá realizar búsqueda de coincidencias de texto en la información de los lugares.	Media
RF14	Búsqueda de lugares cercanos	El sistema permitirá realizar búsqueda de lugares que se encuentren en un determinado radio a partir de la ubicación actual del usuario.	Media
RF15	Llamada telefónica	El sistema permitirá llamar al número de teléfono de un lugar siempre que sean dados los permisos necesarios y el lugar disponga de número.	Alta
RF16	Escritura de correo	El sistema permitirá escribir un correo al lugar seleccionado siempre que el lugar disponga de él.	Alta
RF17	Apertura de dirección web	El sistema permitirá abrir la página web de un lugar en el navegador elegido por el usuario siempre que el lugar disponga de web.	Alta

ID	Nombre	Descripción	Prioridad
RF18	Mostrado en mapa	El sistema permitirá mostrar un lugar en el mapa, centrando la pantalla sobre él.	Alta
RF19	Trazado de ruta a lugar	El sistema permitirá encontrar y trazar la ruta a un lugar siempre que se esté conectado a Internet y se disponga de la ubicación del usuario.	Alta
RF20	Preferencias	El sistema almacenará los siguientes valores definidos por el usuario: tipo de mapa (normal, satélite, terreno, híbrido); visibilidad de monumentos, museos, oficinas de turismo, personalizados; modo de la ruta (a pie, en coche); color de la ruta (rojo, verde, azul, amarillo, blanco, negro); distancia lugares cercanos (500m, 1km, 2km, 3km, 5km)	Alta
RF21	Creación de rutas	El sistema permitirá crear rutas por parte del usuario, no pudiendo tener dos rutas el mismo nombre pero sí los mismos lugares, y contando con ninguno, uno o más lugares de cualquier tipo.	Alta
RF22	Visualizado de rutas	El sistema permitirá mostrar todas las rutas que se encuentren en la base de datos.	Alta
RF23	Visualizado de ruta	El sistema permitirá mostrar los lugares que se encuentren en una ruta.	Alta
RF24	Edición del nombre de ruta	El sistema permitirá cambiar el nombre de una ruta, siempre que éste no sea el mismo que el de otra ruta existente.	Media
RF25	Eliminado de ruta	El sistema permitirá eliminar cualquier ruta.	Alta
RF26	Trazado de ruta	El sistema permitirá encontrar y trazar la ruta entre los lugares agregados a una ruta siempre que se esté conectado a Internet, y entre el primer lugar de la ruta y la ubicación del usuario siempre que tenga permiso y esté activada.	Alta

<b>ID</b>	<b>Nombre</b>	<b>Descripción</b>	<b>Prioridad</b>
RF27	Edición de ruta	El sistema permitirá agregar lugares a la ruta especificada por el usuario, pudiendo cancelar la edición en cualquier momento.	Alta
RF28	Compartición de ruta	El sistema permitirá compartir rutas al usuario, por medio del correo electrónico u otro medio de su elección, incluyendo lugares personalizados si estuvieran agregados.	Media
RF29	Importado de ruta	El sistema permitirá importar rutas al usuario, siempre que se cumplan una serie de condiciones, de un archivo que esté disponible.	Media
RF30	Ruta	El sistema guardará para cada ruta: su nombre y los lugares que contiene.	Alta

### 5.3. Requisitos no funcionales

ID	Nombre	Descripción	Prioridad
RNF01	Implementación de la aplicación	El sistema se implementará como una aplicación para móviles con sistema operativo Android.	Alta
RNF02	Versión mínima de API	La versión mínima de Android para poder usar el sistema será la API 19.	Alta
RNF03	Versión recomendada de API	La versión recomendada de API para usar el sistema será la API 26.	Alta
RNF04	Versión de API para pruebas	Las pruebas se realizarán en dispositivos con niveles de API 19, 21, 23 y 26.	Alta
RNF05	Base de Datos	Se usará SQLite y Android Room para implementar el S.G.B.D.	Alta
RNF06	Formato de los archivos	Se utilizarán los siguientes formatos para los archivos: Monumentos, XML; Museos, CSV, Oficinas, CSV.	Media
RNF07	Proveedor de mapas	El sistema usará el proveedor de mapas de Google.	Alta
RNF08	Proveedor de GPS	El sistema usará el proveedor de GPS de Android, o cualquier alternativa siempre que sea gratuita.	Alta
RNF09	Buscador de rutas	Se usará la API de Google para encontrar rutas entre dos puntos.	Alta
RNF10	Cluster de lugares	El sistema agrupará en clusters los lugares cercanos para no entorpecer la visibilidad al usuario.	Alta
RNF11	Archivo de ruta compartida	El tipo de los archivos de ruta compartida será «.intour»	Media
RNF12	Traducción	Se incorporarán traducciones al español, inglés, francés y alemán.	Media

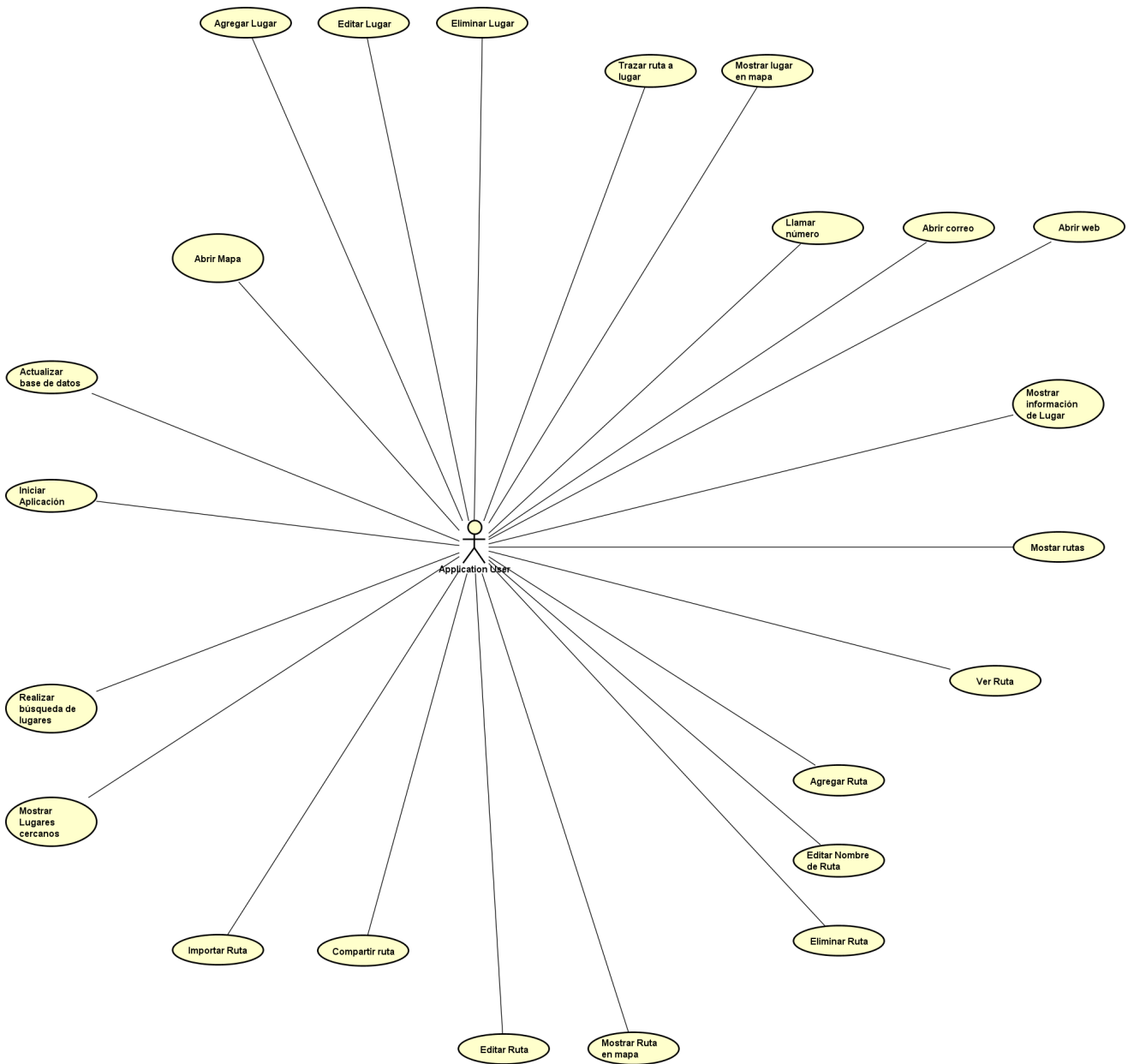
### 5.4. Actores

Sólo se identifica un único actor, el actor Application User:

- **Application User:** usuario de la aplicación, realiza todas las funciones especificadas para la aplicación.

## 5.5. Casos de Uso

Partiendo del análisis, la especificación de requisitos y de actores, se crea el diagrama de casos de uso y a continuación se detalla cada uno de los casos de uso individualmente. (*Todos los diagramas de UML de este trabajo se encuentran en la carpeta imágenes y en el archivo InfoTourist.astah. Más información en la sección Distribución de contenidos del soporte digital*)



**Figura 3:** Diagrama de Casos de Uso

<b>Caso de Uso</b>	Iniciar Aplicación
<b>Dependencias</b>	RF01, RNF01, RNF05, RNF06
<b>Resumen</b>	El usuario inicia la aplicación
<b>Actor</b>	Application User
<b>Pre-condición</b>	La aplicación está instalada en el dispositivo
<b>Post-condición</b>	La aplicación se ha iniciado
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User pulsa sobre el icono de la aplicación</li> <li>2 La aplicación se abre</li> <li>3 La aplicación comprueba si existen datos en el dispositivo</li> <li>4 La aplicación se inicia</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>3.1 Si no existen datos en el dispositivo, se comprueba la conexión a Internet, se deshabilitan las opciones de la pantalla, se descargan del portal de datos abiertos de la junta, habilitan las opciones de la pantalla y el caso de uso continúa en el paso 4</li> <li>3.1.1 Si no se está conectado a Internet, se informa al usuario y el caso de uso queda sin efecto</li> </ol>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Actualizar Base de Datos
<b>Dependencias</b>	RF01, RF02, RF03, RF11, RNF05, RNF06
<b>Resumen</b>	Se actualiza la base de datos con los lugares
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Iniciar Aplicación» y la aplicación cuenta con datos en el dispositivo
<b>Post-condición</b>	Los datos quedan actualizados
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User selecciona la opción «Actualizar base de datos»</li> <li>2 La aplicación comprueba si se está conectado a Internet</li> <li>3 La aplicación deshabilita las opciones de la pantalla</li> <li>4 La aplicación actualiza los datos que se encuentren en el dispositivo</li> <li>5 La aplicación informa al usuario al finalizar la actualización y vuelve a habilitar las opciones</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2.1 Si no se está conectado a Internet, se informa al usuario y el caso de uso queda sin efecto</li> </ol>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Abrir Mapa
<b>Dependencias</b>	RF04, RF05, RF07, RNF05, RNF07, RNF08. RNF10
<b>Resumen</b>	Se abre el mapa y se muestran los lugares
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso “Iniciar Aplicación”
<b>Post-condición</b>	Se muestran los lugares en el mapa
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User selecciona la opción «Iniciar Mapa»</li> <li>2 La aplicación comprueba los permisos de localización</li> <li>3 La aplicación establece los servicios de localización</li> <li>4 La aplicación obtiene la información de los lugares de la Base de Datos</li> <li>5 La aplicación muestra los lugares en el mapa</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2.1 Si no se tienen permisos, la aplicación solicita permisos al usuario y el caso de uso continúa en el paso 2</li> <li>3.1 Si no se han dado permisos, el caso de uso continúa en el paso 4</li> </ol>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Agregar Lugar
<b>Dependencias</b>	RF03, RF04, RF08, RF11, RNF05
<b>Resumen</b>	Se pulsa sobre un lugar del mapa y se agrega un lugar
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa»
<b>Post-condición</b>	
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User mantiene pulsado sobre un sitio del mapa</li> <li>2 La aplicación abre el formulario de edición de lugar</li> <li>3 El actor Application User introduce los datos del lugar que desee (nombre, descripción, etc.)</li> <li>4 El actor Application User confirma los cambios</li> <li>5 La aplicación crea el lugar con los datos introducidos y lo muestra en el mapa en las coordenadas donde se pulsó</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2.1, 3.1, 4.1 Si el actor Application User selecciona la opción «Cancelar», el caso de uso queda sin efecto</li> <li>4.2 Si no se introduce el nombre del lugar, se informa al usuario y el caso de uso vuelve al paso 3</li> </ol>
<b>Secuencia de excepción</b>	



<b>Caso de Uso</b>	Editar lugar
<b>Dependencias</b>	RF04, RF06, RF09, RF11, RNF05
<b>Resumen</b>	Se selecciona el lugar a mostrar, se edita y se persisten los cambios
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar», y el lugar es de tipo «Personalizado»
<b>Post-condición</b>	Se modifican los datos del lugar y se actualiza el mapa
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Editar Lugar»</p> <p>2 La aplicación abre el formulario de edición de lugar, mostrando los valores actuales de ese lugar</p> <p>3 El actor Application User modifica ninguno, uno o varios de los campos del lugar</p> <p>4 El actor Application User confirma los cambios</p> <p>5 La aplicación actualiza los campos del lugar que se hayan modificado y muestra el lugar en el mapa con los datos actualizados</p>
<b>Secuencia alternativa</b>	<p>2.1, 3.1, 4.1 Si el actor Application User selecciona la opción “Cancelar”, el caso de uso queda sin efecto</p> <p>4.2 Si no se introduce el nombre del lugar, se informa al usuario y el caso de uso vuelve al paso 3</p>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Eliminar lugar
<b>Dependencias</b>	RF06, RF10, RNF05
<b>Resumen</b>	Se selecciona el lugar a eliminar, se pulsa sobre la opción y se persiste el cambio
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar», y el lugar es de tipo «Personalizado»
<b>Post-condición</b>	El lugar se elimina de la BD y se actualiza el mapa
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Eliminar Lugar»</p> <p>2 La aplicación elimina el lugar de la Base de Datos y actualiza el mapa</p>
<b>Secuencia alternativa</b>	2.1 Si el lugar está agregado a una ruta, se muestra un mensaje informando al usuario y el caso de uso queda sin efecto
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Trazar ruta a lugar
<b>Dependencias</b>	RF05, RF19, RF20, RNF09
<b>Resumen</b>	Se pulsa sobre el lugar deseado, se selecciona la opción de trazar ruta y se muestra la ruta encontrada
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa» o «Mostrar información de Lugar»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Llévame hasta aquí»</p> <p>2 La aplicación comprueba la conexión a Internet</p> <p>3 La aplicación comprueba si se dispone de la localización GPS del usuario</p> <p>4 La aplicación busca la ruta entre la localización del usuario y el lugar seleccionado</p> <p>5 La aplicación dibuja en el mapa la ruta encontrada con el color definido por el usuario y mueve la cámara a la posición del actor</p>
<b>Secuencia alternativa</b>	<p>1.1 Si ya hay una ruta dibujada en el mapa, se borra y el caso de uso continúa en el paso 2</p> <p>2.1 Si no se está conectado a Internet, se informa al usuario y el caso de uso queda sin efecto</p> <p>3.1 Si no se dispone de la localización GPS del usuario, se informa al usuario y el caso de uso queda sin efecto</p> <p>5.1 Si no se encuentra ninguna ruta, se informa al usuario y el caso de uso queda sin efecto</p>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Mostrar lugar en mapa
<b>Dependencias</b>	RF04, RF05, RF18
<b>Resumen</b>	Se selecciona la opción «Mostrar en Mapa» y se muestra el lugar en el mapa
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar»
<b>Post-condición</b>	Se muestra el lugar en el centro de la pantalla
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Mostrar en mapa»</p> <p>2 La aplicación centra en la pantalla el lugar seleccionado</p>
<b>Secuencia alternativa</b>	
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Llamar número
<b>Dependencias</b>	RF06, RF11, RF15
<b>Resumen</b>	Se llama al número de teléfono del lugar
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar»
<b>Post-condición</b>	
<b>Secuencia base</b>	1 El actor Application User selecciona la opción «Llamar teléfono» 2 La aplicación comprueba los permisos de llamadas 3 La aplicación llama al número de teléfono
<b>Secuencia alternativa</b>	Si no se tienen permisos, la aplicación solicita permisos al usuario y el caso de uso continúa en el paso 2 Si no se han dado permisos, el caso de uso queda sin efecto
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Abrir correo
<b>Dependencias</b>	RF06, RF11, RF16
<b>Resumen</b>	Se abre un nuevo correo con destinatario el correo del lugar
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar»
<b>Post-condición</b>	
<b>Secuencia base</b>	1 El actor Application User selecciona la opción «Abrir correo» 2 La aplicación abre un nuevo correo con destinatario el encontrado en el lugar
<b>Secuencia alternativa</b>	
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Abrir web
<b>Dependencias</b>	RF06, RF11, RF17
<b>Resumen</b>	Se abre la página web del lugar en el navegador elegido por el usuario
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar información de Lugar»
<b>Post-condición</b>	
<b>Secuencia base</b>	1 El actor Application User selecciona la opción «Abrir Web» 2 La aplicación abre el navegador con la web del lugar
<b>Secuencia alternativa</b>	
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Mostrar información de lugar
<b>Dependencias</b>	RF06, RF11, RF12
<b>Resumen</b>	Se pulsa sobre el lugar deseado y se muestra la información de ese lugar
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa»
<b>Post-condición</b>	Se muestra correctamente la información del lugar
<b>Secuencia base</b>	1 El actor Application User selecciona el lugar a mostrar y pulsa sobre la ventana de información 2 La aplicación muestra la información del lugar, completando los campos sin información con el valor «No disponible»
<b>Secuencia alternativa</b>	1.1 Se ejecuta el caso de uso «Mostrar lugares cercanos» y el actor Application User selecciona uno de los lugares 1.2 Se ejecuta el caso de uso «Realizar búsqueda de lugares» y el actor Application User selecciona uno de los lugares 1.3 Se ejecuta el caso de uso «Ver ruta» y el actor Application User selecciona uno de los lugares
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Mostrar Rutas
<b>Dependencias</b>	RF22, RF30, RNF05
<b>Resumen</b>	Se muestran los lugares de una ruta
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa»
<b>Post-condición</b>	
<b>Secuencia base</b>	1 El actor Application User selecciona la opción «Mostrar Rutas» 2 La aplicación obtiene las rutas de la Base de Datos 3 La aplicación muestra las rutas
<b>Secuencia alternativa</b>	3.1 Si la Base de Datos no contiene rutas, se informa al usuario y el caso de uso continúa
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Agregar Ruta
<b>Dependencias</b>	RF21, RNF06, RF30
<b>Resumen</b>	Se agrega una nueva ruta
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	Se agrega la ruta a la Base de Datos y se actualizan las rutas mostradas
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User selecciona la opción «Agregar Ruta»</li> <li>2 La aplicación muestra el formulario de edición de nombre de ruta</li> <li>3 El actor introduce el nombre de la ruta</li> <li>4 El actor confirma los cambios</li> <li>5 La aplicación añade la ruta a la Base de Datos y actualiza las rutas mostradas</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2.1, 3.1, 4.1 Si el actor selecciona la opción «Cancelar», el caso de uso queda sin efecto</li> <li>5.1 Si ya existe una ruta con el nombre introducido, la aplicación informa al usuario y el caso de uso vuelve al paso 2</li> <li>5.2 Si no se ha introducido el nombre de la ruta, la aplicación informa al usuario y el caso de uso vuelve al paso 2</li> </ol>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Editar ruta
<b>Dependencias</b>	RF05, RF03, RF27, RF30, RNF06
<b>Resumen</b>	Se selecciona la opción «Editar ruta», se edita el nombre y se persiste
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El actor Application User selecciona la opción «Editar Ruta»</li> <li>2 La aplicación muestra el mapa y activa el modo de edición de ruta</li> <li>3 El actor selecciona ninguno, uno o más lugares para añadirse a la ruta</li> <li>4 El actor confirma los cambios en la ruta</li> <li>5 La aplicación actualiza la ruta, persiste los cambios en la Base de Datos y desactiva el modo de edición de ruta</li> </ol>
<b>Secuencia alternativa</b>	2.1, 3.1, 4.1 Si el actor selecciona la opción cancelar, la aplicación no registra los cambios y el caso de uso queda sin efecto
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Mostrar ruta en mapa
<b>Dependencias</b>	RF26, RF07, RNF08, RNF09
<b>Resumen</b>	Se selecciona la opción mostrar ruta, y se muestra en el mapa
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Mostrar ruta en mapa»</p> <p>2 La aplicación busca la ruta entre la localización actual y los lugares que contiene la ruta</p> <p>3 La aplicación muestra en el mapa la ruta, con el color definido por el usuario</p>
<b>Secuencia alternativa</b>	<p>2.1 Si no se está conectado a Internet, se informa al usuario y el caso de uso queda sin efecto</p> <p>2.2 Si la aplicación no encuentra la ruta, se informa al usuario y el caso de uso queda sin efecto</p> <p>3.1 Si ya hay una ruta dibujada, se borra y el caso de uso continúa</p>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Eliminar ruta
<b>Dependencias</b>	RF25, RNF06
<b>Resumen</b>	Se selecciona la opción «Eliminar ruta», y se persiste
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	Se elimina la ruta y se actualizan las rutas a mostrar
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Eliminar Ruta»</p> <p>2 La aplicación solicita confirmación al usuario</p> <p>3 El actor confirma</p> <p>4 La aplicación elimina los itinerarios de la ruta y la ruta</p> <p>5 La aplicación actualiza las rutas a mostrar</p>
<b>Secuencia alternativa</b>	3.1 Si el actor no confirma los cambios, el caso de uso queda sin efecto
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Compartir ruta
<b>Dependencias</b>	RF28, RF30, RNF06, RNF11
<b>Resumen</b>	Se selecciona la opción «Compartir ruta», se escoge la ruta y la opción de compartición y se comparte
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1 El usuario selecciona la opción «Compartir ruta»</li> <li>2 La aplicación comprueba los permisos de escritura en almacenamiento externo</li> <li>3 La aplicación comprueba si está instalado el almacenamiento externo</li> <li>4 La aplicación escribe en un fichero la ruta, incluyendo lugares personalizados si los hubiera</li> <li>5 La aplicación adjunta el fichero al medio elegido para compartir la ruta</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2.1 Si no se han concedido permisos, la aplicación los solicita</li> <li>3.1 Si no se han concedido permisos, se informa al usuario y el caso de uso queda sin efecto</li> <li>3.2 Si no se ha instalado el almacenamiento externo, se informa al usuario y el caso de uso queda sin efecto</li> <li>4.1 Si no se ha podido escribir el fichero, se informa al usuario y el caso de uso queda sin efecto</li> </ol>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Importar ruta
<b>Dependencias</b>	RF29, RF30, RNF06, RNF11
<b>Resumen</b>	Se selecciona la opción «Importar ruta», se escoge la ruta seleccionada y se persiste
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Importar ruta»</p> <p>2 La aplicación comprueba si se tienen permisos de lectura del almacenamiento externo</p> <p>3 La aplicación abre un selector de archivos</p> <p>4 El actor selecciona un archivo</p> <p>5 La aplicación comprueba la dirección del archivo</p> <p>6 La aplicación comprueba que el nombre del archivo siga un formato específico (TEXTO-SEPARADOR-NUMERO)</p> <p>7 La aplicación comprueba que no exista una ruta con el nombre de la ruta a importar</p> <p>8 La aplicación comprueba que el archivo es un fichero y de tipo «.intour»</p> <p>9 La aplicación lee la información contenida en el fichero y crea la ruta, su itinerario y los lugares personalizados si los hubiera, informando al usuario al finalizar</p>
<b>Secuencia alternativa</b>	<p>2.1 Si no se tienen permisos, la aplicación los solicita y el caso de uso vuelve al paso 2</p> <p>3.1 Si no se han dado permisos, la aplicación informa al usuario y el caso de uso queda sin efecto</p> <p>4.1 Si el usuario selecciona la opción «Retrosceso», el caso de uso queda sin efecto</p> <p>5.1 Si la aplicación no es capaz de encontrar la dirección, se informa al usuario y el caso de uso queda sin efecto</p> <p>6.1 Si el nombre del archivo no sigue el formato, la aplicación informa al usuario y el caso de uso queda sin efecto</p> <p>7.1 Si ya existe una ruta con el nombre en la Base de Datos, la aplicación informa al usuario y el caso de uso queda sin efecto</p> <p>8.1 Si el archivo no es un fichero, la aplicación informa al usuario y el caso de uso queda sin efecto</p> <p>8.2 Si el fichero no es de tipo «.intour», la aplicación informa al usuario y el caso de uso queda sin efecto</p> <p>9.1 Si se produce algún fallo durante la lectura del fichero, la aplicación informa al usuario y el caso de uso queda sin efecto</p>
<b>Secuencia de excepción</b>	



<b>Caso de Uso</b>	Editar nombre de ruta
<b>Dependencias</b>	RF24, RF30, RNF06
<b>Resumen</b>	Se modifica el nombre de una ruta
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas» y existe al menos una ruta
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Cambiar Nombre»</p> <p>2 La aplicación muestra el formulario de edición de nombre de ruta</p> <p>3 El actor introduce un nuevo nombre de ruta</p> <p>4 El actor confirma los cambios</p> <p>5 La aplicación modifica el nombre de la ruta y actualiza las rutas que se muestran</p>
<b>Secuencia alternativa</b>	<p>2.1, 3.1, 4.1 Si el actor selecciona la opción “Cancelar”, el caso de uso queda sin efecto</p> <p>5.1 Si ya existe una ruta con el nombre introducido, la aplicación informa al usuario y el caso de uso vuelve al paso 2</p> <p>5.2 Si no se ha introducido el nombre de la ruta, la aplicación informa al usuario y el caso de uso vuelve al paso 2</p>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Ver ruta
<b>Dependencias</b>	RF23, RF30, RNF06
<b>Resumen</b>	Se muestran los lugares de una ruta
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Mostrar Rutas»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Ver ruta»</p> <p>2 La aplicación obtiene el itinerario de la ruta seleccionada</p> <p>3 La aplicación muestra los lugares que contiene la ruta</p>
<b>Secuencia alternativa</b>	2.1 Si la ruta no contiene lugares, se informa al usuario y el caso de uso queda sin efecto
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Mostrar lugares cercanos
<b>Dependencias</b>	RF07, RF14, RF20, RNF06
<b>Resumen</b>	Se selecciona la opción mostrar lugares cercanos y se muestran
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Mostrar Lugares cercanos»</p> <p>2 La aplicación comprueba si está activada la localización</p> <p>3 La aplicación busca los lugares que se encuentren a menor o igual distancia que el valor definido en las preferencias</p> <p>4 La aplicación muestra los lugares encontrados</p>
<b>Secuencia alternativa</b>	<p>2.1 Si no está activada la localización, se informa al usuario y el caso de uso queda sin efecto</p> <p>4.1 Si no se encuentran lugares, o se encuentran pero no están visibles, se informa al usuario y el caso de uso queda sin efecto</p>
<b>Secuencia de excepción</b>	

<b>Caso de Uso</b>	Realizar búsqueda de lugares
<b>Dependencias</b>	RF13, RFN06
<b>Resumen</b>	Se selecciona la opción «Búsqueda», se introducen los términos a buscar y se muestran los resultados
<b>Actor</b>	Application User
<b>Pre-condición</b>	Se ha ejecutado el caso de uso «Abrir Mapa»
<b>Post-condición</b>	
<b>Secuencia base</b>	<p>1 El actor Application User selecciona la opción «Buscar»</p> <p>2 La aplicación abre el buscador</p> <p>3 El actor introduce el texto a buscar</p> <p>4 La aplicación busca coincidencias del texto en varios campos del lugar (nombre, tipo, provincia, descripción)</p> <p>5 La aplicación muestra los lugares que contienen coincidencias de texto</p>
<b>Secuencia alternativa</b>	<p>2.1, 3.1 Si el actor selecciona la opción “Retroseso”, el caso de uso queda sin efecto</p> <p>5.1 Si no se encuentran coincidencias, se informa al usuario y el caso de uso queda sin efecto</p> <p>5.2 Si un lugar contiene coincidencias pero no está habilitada su visibilidad, no se muestra el lugar y el caso de uso continúa</p>
<b>Secuencia de excepción</b>	

## 5.6. Modelo de Dominio

A partir de todo lo anteriormente expuesto en esta sección, se detalla el modelo de dominio de la aplicación en análisis.

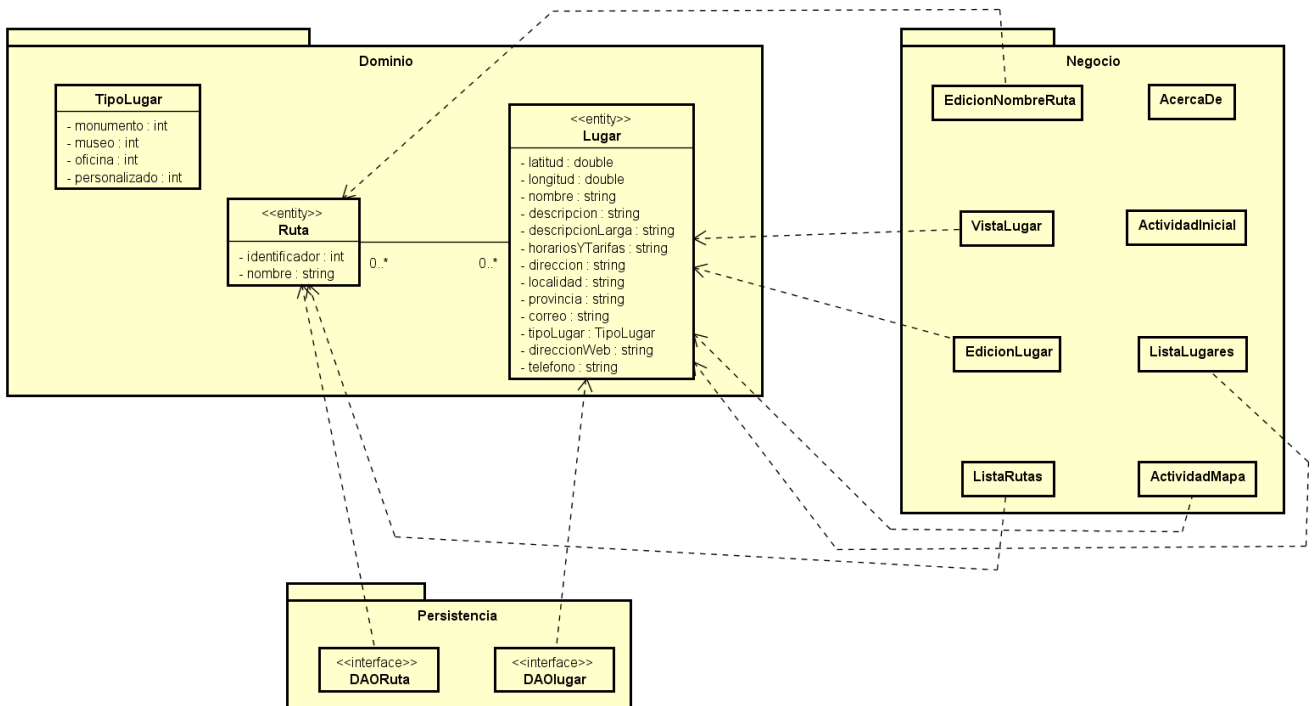


Figura 4: Diagrama de Clases en análisis

Se decide dividir la aplicación en tres paquetes:

- **Dominio:** en este paquete se incluyen las clases del dominio de la aplicación. Se encuentran dos clases, Ruta y Lugar y una tercera clase auxiliar, TipoLugar.
- **Negocio:** en este paquete se encuentran las clases que representan las actividades de Android, que realizarán la parte de negocio de la aplicación.
- **Persistencia:** En este paquete se hallan las clases que hacen el acceso a la base de datos. Se sigue el patrón DAO (Data Access Object). Hay una por cada clase del Dominio.

No se decide incluir el paquete «Presentación» al ser dependiente de la arquitectura de Android.

### Dominio

En este paquete se incluyen las clases Lugar, Ruta y TipoLugar. La clase Lugar representa a un lugar de la aplicación, del que se guarda la siguiente información:

- Latitud

- Longitud (estos dos valores representan las coordenadas del lugar, son necesarios para poder posicionar el marcador del lugar en el mapa)
- Nombre
- Descripción (utilizada en el caso de los monumentos para refinar aún más su tipo, al poder ser palacios, castillos, iglesias, etc)
- Descripción Larga (descripción propiamente dicha)
- Horarios y Tarifas
- Dirección (nombre de calle y número)
- Localidad
- Provincia
- Correo (correo electrónico)
- TipoLugar (el tipo del lugar)
- Dirección Web
- Teléfono (a pesar de ser un número se usa un String, ya que en los ficheros de la Junta los teléfonos son guardados de esta forma y una gran cantidad de ellos no están formateados para ser admitidos en un entero)

Esta será la clase usada para representar cada uno de los lugares encontrados en los ficheros de la Junta.

La clase TipoLugar es un enumerado utilizado para representar el tipo de cada lugar. Consta de cuatro tipos: monumento, museo, oficina (de información turística) y personalizado (lugares definidos por el usuario de la aplicación). Esta clase es usada por Lugar.

La clase Ruta se usa para representar las rutas que puede definir el usuario. Sus atributos son

- Identificador
- Nombre

Para representar el itinerario de una ruta se utiliza una relación con multiplicidad 0..\* (cero a muchos) entre Lugar y Ruta. De esta forma se entiende que, de existir una Ruta, puede tener agregados ningún lugar, uno o varios; es decir, se permite la existencia de rutas vacías (sin lugares agregados) ya que, al crear una Ruta, no tiene todavía ningún lugar agregado. En el otro sentido de la navegabilidad un Lugar puede no estar agregado a ninguna Ruta, estar agregado a una o a varias.

Tanto la clase Ruta como la clase Lugar son *Entities*, ya que se considera que esta es la mejor opción para tratar con los datos que se encuentran en la base de datos, ser la que propone Android al trabajar con Room[12][13] y resultar considerablemente más cómodo trabajar con este paradigma.

## Negocio

En este paquete se encuentran las clases que representan a la parte de negocio de la aplicación. Al estar en fase de análisis su comportamiento no está especificado de manera detallada, por lo que sólo se incluye en líneas generales. (*En la sección Implementación-Casos de Uso se detalla este comportamiento*). Todos los casos de uso quedan cubiertos por las clases de este paquete.

- EdicionNombreRuta: se encarga de la edición del nombre de una Ruta, tanto si es de una ruta ya existente como si es una nueva ruta, persistiendo el cambio o informando al usuario en caso contrario. Depende de Ruta.
- AcercaDe: muestra información sobre la aplicación (esta clase se añadió en una versión tardía de la fase de análisis)
- VistaLugar: muestra la información de un lugar especificada en los apartados anteriores. Permite además mostrar ese lugar en el mapa, trazar la ruta al lugar, agregar el lugar a una ruta si está el modo de edición activado, eliminar un lugar si es de tipo «Personalizado», llamar al teléfono, escribir un correo y abrir la página web, en caso de existir. Depende de Lugar.
- ActividadInicial: encargada de descargar los datos en caso de ser el primer inicio de la aplicación y actualizar los datos cuando decida el usuario.
- EdicionLugar: clase cuyo cometido es editar un lugar, en los casos de edición de uno existente y se esté editando su información y añadido de un nuevo lugar. Como se ha expuesto anteriormente sólo se permiten estas operaciones cuando el tipo del Lugar es «Personalizado». Depende de Lugar.
- ListaLugares: clase que muestra una lista de lugares, pudiendo elegir cualquiera de ellos. Esta lista se muestra cuando se buscan coincidencias de texto, se muestran los lugares cercanos o se ven los lugares agregados a una ruta. Depende de Lugar.
- ListaRutas: clase que muestra una lista con las rutas de la aplicación. Desde esta clase se pueden realizar varias operaciones con las rutas: ver su itinerario, mostrar en el mapa, cambiar su nombre, editar la ruta, eliminarla y compartirla. Además se puede agregar una nueva ruta e importarla desde un fichero. Depende de Ruta.
- ActividadMapa: clase que muestra en el mapa los lugares. Permite agregar nuevos lugares de tipo «Personalizado», trazar la ruta a un lugar, establecer los servicios de localización,

buscar coincidencias de texto, buscar los lugares cercanos. En el modo de edición permite agregar lugares a la ruta seleccionada. Depende de Lugar.

Exceptuando AcercaDe, ListaLugares, EdicionLugar, EdicionNombreRuta, las demás clases permiten acceder a las preferencias de la aplicación y lanzar la actividad AcercaDe.

## **Persistencia**

En este paquete se hallan las clases que interactúan con la base de datos. Se sigue el patrón *Data Access Object* [11] al ser el recomendado por Android [6]. De esta forma, cada una de las clases del dominio (Ruta y Lugar) cuentan con su DAO, en este caso DAORuta y DAOLugar, que se encargan de todas las operaciones de acceso a datos *CRUD* (Create, Read, Update, Delete) [10]. Este comportamiento es idéntico para cada DAO, dependiendo DAORuta de Ruta y DAOLugar de Lugar. Toda clase que necesite realizar operaciones sobre estas clases, debe llamar al método adecuado del DAO.

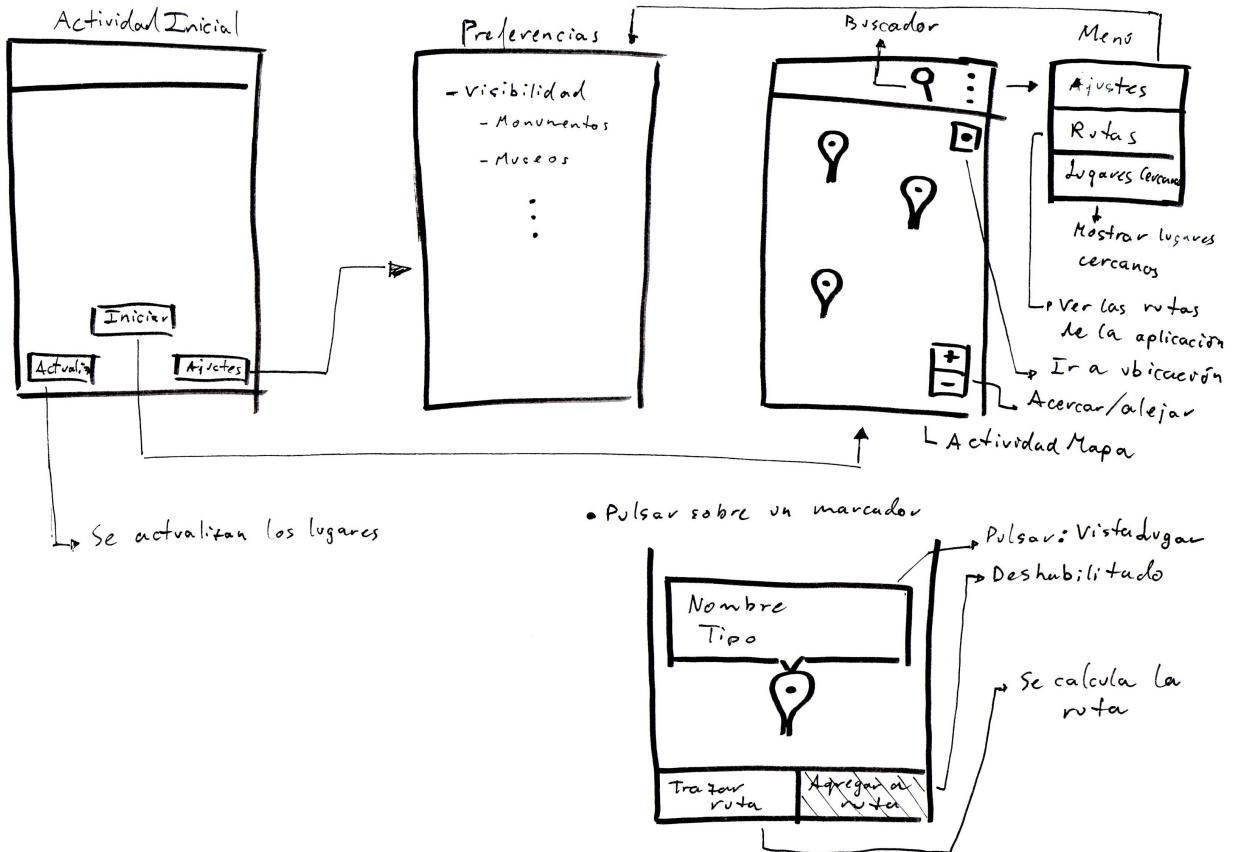
Todos estos patrones de diseño se eligen de acuerdo con lo visto a lo largo de la carrera (Asignaturas de Modelado de Software y Diseño de Software)

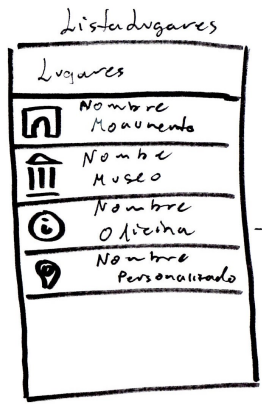
# 6. Implementación

## 6.1. Prototipo en papel

Se realiza una aproximación a la aplicación en forma de un prototipo en papel, que replica el comportamiento deseado. De esta forma se comprueba la usabilidad de la aplicación antes de comenzar a programar.

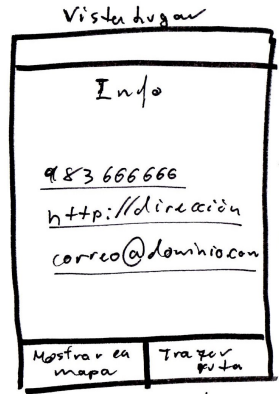
4





- se abre al tener que mostrar una lista de lugares

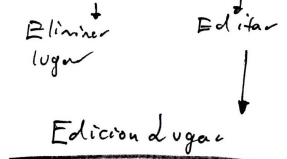
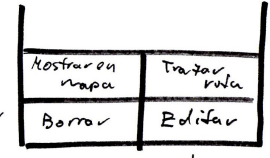
pulsar sobre un lugar



Subrayados y en azul

Se mueve la cámara al lugar  
Se calcula la ruta

\* Si es Personalizado



Si se acepta se agrega/actualiza

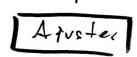
Acerca De



Mantener pulsado sobre la pantalla del mapa

Lista Rutas

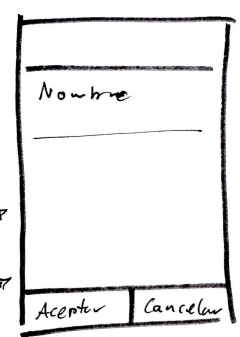
va a ajustes +



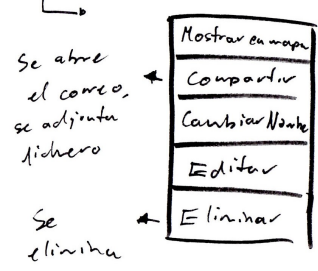
pulsar, se abre lista de lugares

Se abre el selector de ficheros

Edición Nombre Ruta



pulsación carga, se abre menú

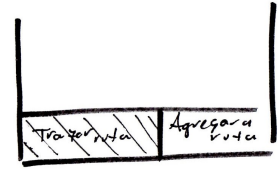


Se abre el correo, se adjunta fichero  
Se elimina

Se dibuja en el mapa

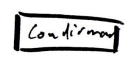
Se activa el "modo edición"

Actividad Mapa



Deshabilitar

- Nueva opción de menú

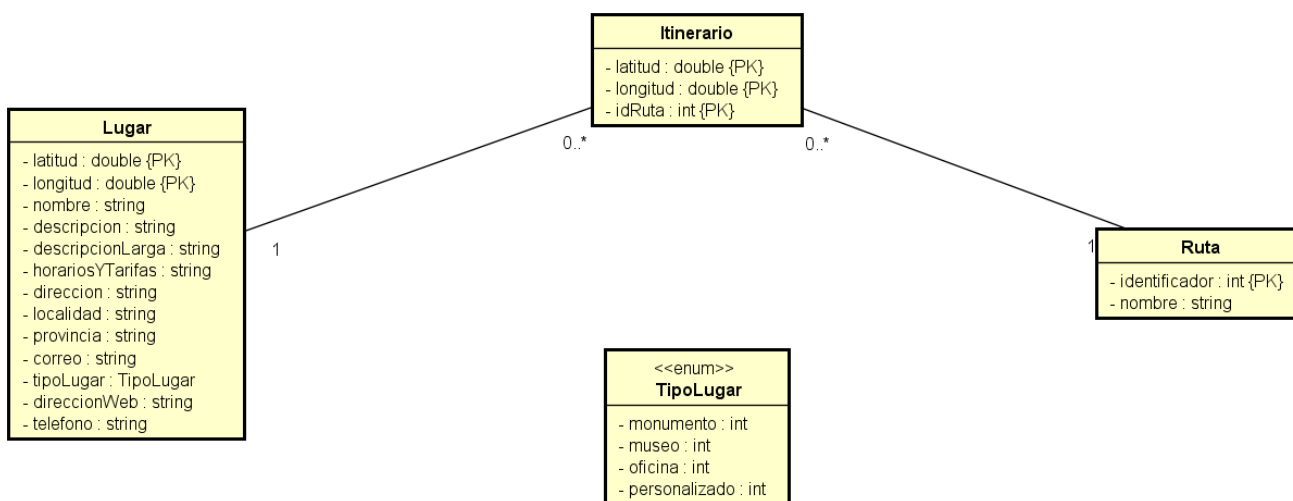




Una de las principales modificaciones respecto a este prototipo fue cambiar la mayoría de los botones que aparecen en las actividades por opciones de menú, y cambiar la forma en la que se muestran los botones «Llévame hasta aquí» y «Agregar a Ruta». Además, por recomendación del tutor se cambia el color de la barra de la aplicación cuando se activa el modo de edición, para que el usuario sepa mejor en qué modo la está usando. Otro cambio fue el de añadir realimentación en bastantes acciones en forma de Snackbars y Toasts.

## 6.2. Modelo de Datos

Se usa SQLite con Android Room como S.G.B.D., por lo que el esquema de datos a seguir será relacional[15]. El modelo de datos queda entonces de la siguiente forma:



**Figura 5:** Modelo de datos

Las tablas Lugar y Ruta mantienen las mismas características que las respectivas clases del modelo de dominio, especificando como clave primaria para Lugar la combinación de los campos latitud y longitud; y para Ruta el campo identificador. La tabla TipoLugar es un enumerado que representa de igual forma a TipoLugar del modelo de dominio.

Para poder representar la relación entre Lugar y Ruta con multiplicidad 0..\* en los dos extremos se utiliza una tabla intermedia llamada Itinerario, con tres claves foráneas (latitud, longitud referencian a los mismos campos de Lugar, idRuta referencia al identificador de Ruta) y como clave primaria la combinación de estos tres valores. Itinerario tiene una relación a 1 con Ruta y con Lugar, mientras que estas dos tablas tienen la multiplicidad 0..\*. De esta forma se puede identificar unívocamente el itinerario de una Ruta, al poder navegar desde una ruta determinada a todos sus lugares, y en concreto al lugar que se desee. Por otra parte, desde un Lugar podemos obtener las rutas a las que esté agregado. Como un Lugar sólo puede aparecer una vez dentro de una Ruta, la clave primaria (latitud, longitud, idRuta) es válida e identifica de forma correcta los

itinerarios, es decir, nunca se va a poder repetir.

Al usarse Android Room no fue necesario escribir ningún código SQL, ya que se proporcionan las anotaciones necesarias para transformar las clases en entities, especificando relaciones y nombre de los campos de las tablas. A mayores, hubo que cambiar el nombre de alguno de los campos para que se introdujeran con el nombre adecuado en la base de datos, por ejemplo, idRuta se modificó para que se llamara id\_ruta en la base de datos.

Todo lo relacionado con esta sección se ha visto en la asignatura *Análisis y Diseño de Bases de Datos*.

### 6.3. Modelo de Dominio

En la fase de implementación, se modifica el modelo de dominio de la siguiente forma, para reflejar los cambios introducidos por la implementación en Android, la necesaria creación de clases auxiliares y reflejar los cambios introducidos por el modelo de datos de la base de datos. A continuación se explica el contenido de cada uno de los paquetes. *(Nota: se puede encontrar el diagrama completo del modelo de dominio en el archivo astah y en la carpeta imágenes)*

#### 6.3.1. Dominio

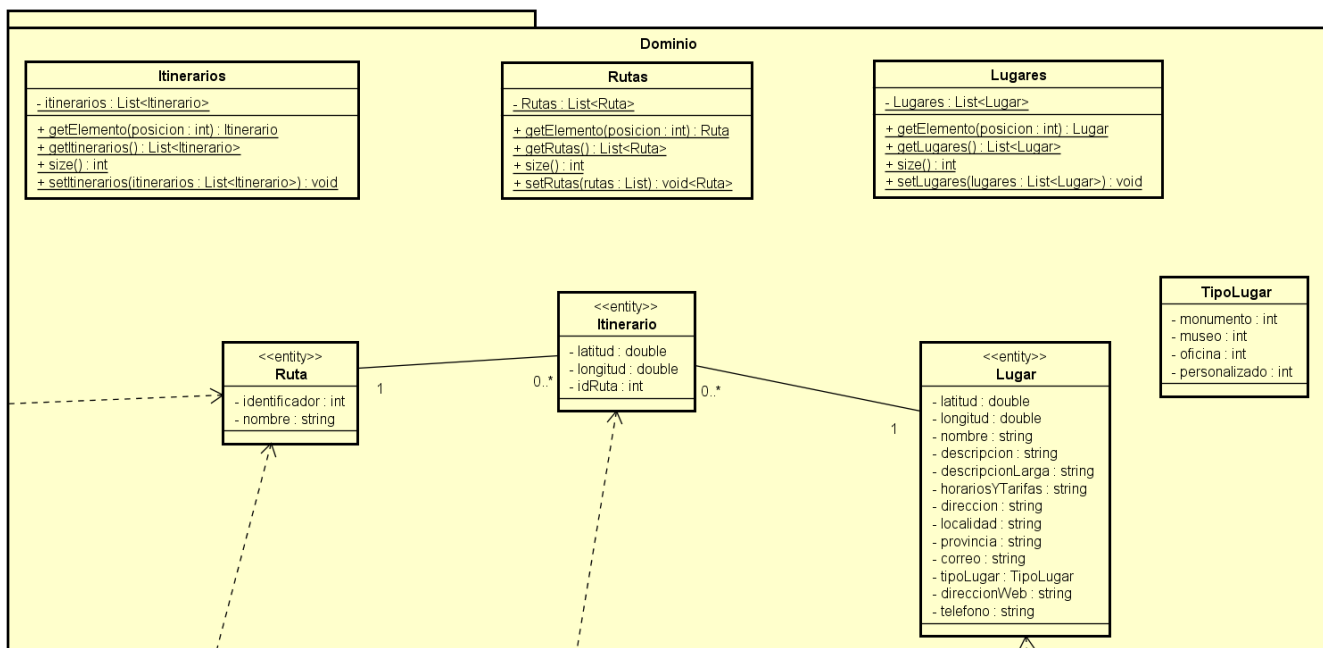


Figura 6: Paquete Dominio

Las clases Ruta y Lugar permanecen igual que en análisis, pero se introduce una nueva clase Itinerario para reflejar el modelo de datos, manteniendo la misma multiplicidad de este modelo.



Room para almacenar tipos de datos personalizados en la base de datos[18].

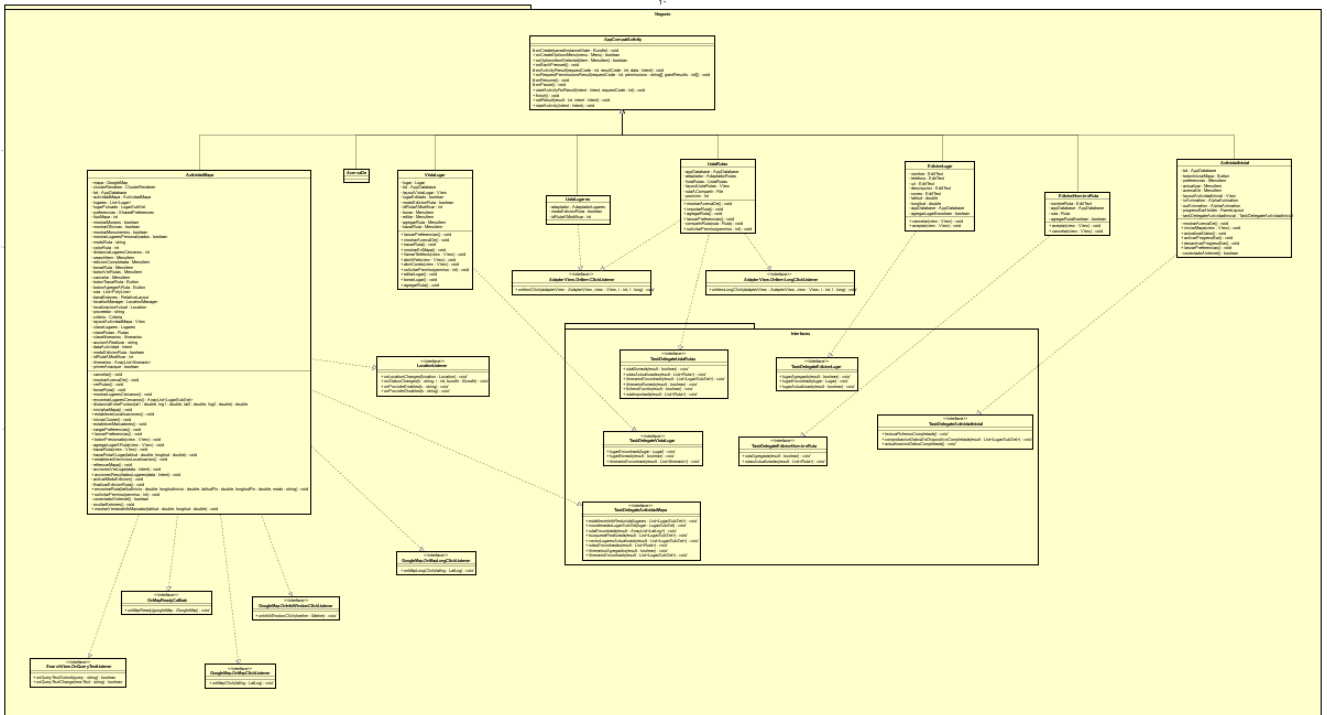
Como se verá más adelante, la mayoría de las operaciones que tienen que ver con lugares (mostrar en el mapa, agregar a ruta, trazar ruta, etc) no necesitan toda la información de la clase Lugar. Ya que se encuentran bastantes lugares en la base de datos, cargar toda la información en memoria para luego no usarse no resulta adecuado, por lo que se define un subtipo de lugar llamado LugarSubSet, que sólo contiene parte de la información de Lugar (latitud, longitud, nombre, tipoLugar, localidad)[5]. Esta clase será la preferida a la hora de realizar la mayor parte de las operaciones sobre lugar por ser más ligera que la clase Lugar.

En Android, el acceso a datos se debe realizar en un hilo de ejecución en segundo plano diferente al principal, para no bloquear la interfaz y permitir al usuario seguir operando con la aplicación. Se dan dos alternativas que permiten este comportamiento: Thread y AsyncTask. Esta última[8] incorpora una serie de operaciones que resultan de interés, en concreto onPostExecute, que permite comunicar al hilo principal la finalización de la ejecución en segundo plano. Se decide usar AsyncTask porque se necesita comunicar el resultado de las operaciones en la mayoría de los casos (por ejemplo, al agregar una ruta hay que actualizar las rutas mostradas una vez que ha sido agregada), y sólo es posible con el método de onPostExecute, ya que con Thread no sabemos cuándo ha terminado de ejecutarse el hilo. Por lo tanto el esquema a seguir a la hora de operar con la base de datos será el siguiente:

1. Se define una AsyncTask que llama a una operación en concreto de uno de los DAO de la aplicación (por ejemplo AgregarRuta)
2. Desde la actividad se crea la AsyncTask y se ejecuta en segundo plano, se puede seguir usando la aplicación
3. Dentro de la AsyncTask se ejecuta el método doInBackground, donde se realiza el trabajo deseado (en el ejemplo, agregar una ruta)
4. Una vez finalizado se ejecuta el método onPostExecute, que comunica a la actividad correspondiente el resultado de la operación (en el ejemplo, se comunica que se ha agregado la ruta y se pide a la actividad que actualice las rutas que se muestran)
5. La actividad realiza la acción correspondiente (Más adelante se especifica cómo se hace exactamente)

De esta forma se realiza el acceso a datos no bloqueante. Debido a ello se añade un subpaquete (Facades) donde se almacenan las distintas AsyncTask que llaman a los DAO de Lugar, Ruta e Itinerario.

### 6.3.3. Negocio



**Figura 8:** Paquete Negocio

Las clases principales de este paquete son las mismas que las que se especificó en la fase de análisis. Estas clases son Activities de Android, usadas para representar la funcionalidad de la aplicación, por lo que todas heredan de la clase AppCompatActivity, que especifica el comportamiento y el flujo comunes a todas las actividades. Se añaden las distintas operaciones necesarias para implementar los casos de uso. Algunas clases implementan varias interfaces para poder agregar la funcionalidad requerida:

- **ActividadMapa:** implementa `SearchView.OnQueryTextListener` para detectar el texto introducido en el buscador al realizar búsquedas de coincidencias; `OnMapReadyCallback` para conocer cuándo está cargado el mapa en el dispositivo y se puede empezar a utilizar; `GoogleMap.OnMapClickListener` para averiguar las coordenadas cuando el usuario pulsa sobre una posición del mapa, `GoogleMap.OnInfoWindowClickListener` para incorporar funcionalidad al pulsarse sobre la ventana de información de un marcador; `GoogleMap.OnMapLongClickListener` para averiguar las coordenadas cuando el usuario hace una «pulsación larga» sobre una posición del mapa (se abre la actividad de edición de lugar y se usan las coordenadas para agregar el nuevo lugar); `LocationListener` para establecer los servicios de localización y recibir actualizaciones cuando el dispositivo cambia de posición.
- **ListaLugares:** implementa `AdapterView.OnItemClickListener`. Al mostrar esta actividad una lista de lugares, es necesario añadir un listener para detectar cuando el usuario pulsa sobre un elemento de la lista.
- **ListaRutas:** implementa dos interfaces, `AdapterView.OnItemClickListener` y `AdapterView.OnItemLongClickListener`. Esta actividad muestra una lista de rutas, y es

necesario añadir listeners para conocer cuando el usuario pulsa sobre un elemento de la lista y cuando hace una «pulsación larga», es decir, cuando el usuario mantiene pulsado sobre un elemento (en este caso se muestra un menú de acciones que se pueden realizar sobre la ruta)

Como se explica en la sección anterior, una vez que se termina de realizar una operación contra la base de datos, la AsyncTask en ejecución llama a la actividad correspondiente y le notifica su finalización, llamando al método que sea necesario. Para lograr esta funcionalidad existen varias opciones. Entre ellas se encuentra pasar como parámetro del constructor de la AsyncTask la actividad que define el método que debe ser llamado, y asignarlo a un atributo privado. Pero en este caso se crea una fuga de contexto (*Context Leak*[2]), lo cual puede dar lugar a errores y situaciones potencialmente peligrosas. Para resolverlo se crea una interfaz por cada Actividad que necesita ser llamada dentro de las AsyncTasks (en la aplicación, todas menos ListaRutas y AcercaDe), que a su vez es implementada por la actividad correspondiente. Cada una de las interfaces definirá los métodos que se llaman al finalizar las AsyncTasks. El constructor de la AsyncTask recibe como parámetro una instancia de la interfaz (en este caso, la actividad), y llama al método adecuado de la interfaz, que a su vez es redefinido dentro de la actividad. De esta forma, a pesar de tener que definir bastantes interfaces, se soluciona el problema de las fugas de contexto. Se eligió esta opción por ser sencilla y elegante.

En el subpaquete interfaces se encuentran todas estas interfaces, a las que se ha decidido llamar *TaskDelegate* y el nombre de la actividad.

#### 6.3.4. Preferencias

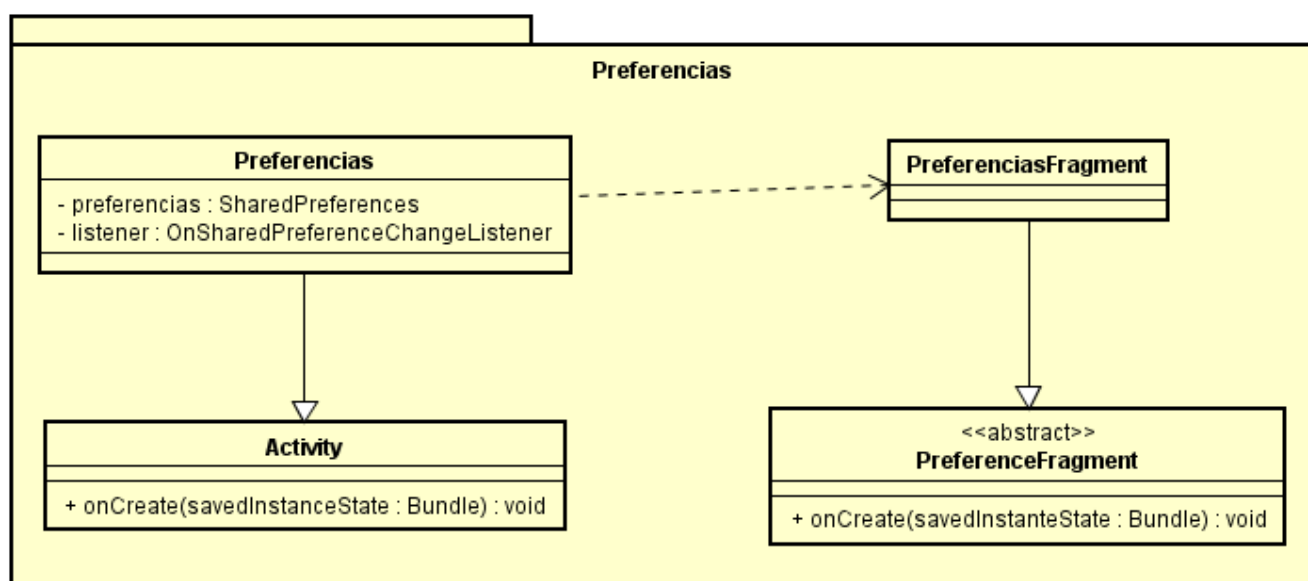


Figura 9: Paquete Preferencias

En este paquete se localizan las clases utilizadas para gestionar las preferencias de usuario, que



- ClusterRenderer y MarcadorCluster: clases utilizadas para poder agrupar los marcadores en clusters, especifican su comportamiento y las acciones a realizar cuando se interactúa con ellos.
- BuscadorRuta y GeneradorRuta: clases que se encargan de obtener y traducir la ruta entre dos lugares dados cuando se llama a la API de Google, para poder mostrarla sobre el mapa en la aplicación.
- AdaptadorLugares y AdaptadorRutas: heredan de BaseAdapter, se utiliza para personalizar un elemento de una ListView en Android (por ejemplo, el icono, el título, la descripción, etc).
- LectorRuta: clase que lee el fichero que contiene la ruta para importar.
- EscritorRuta: clase que escribe el fichero con la ruta para compartir.
- ParseadorFicheros: clase que lee los ficheros de la junta y almacena los lugares que se encuentran en ellos en la base de datos.
- ManejadorXMLLugar y ManejadorXMLRuta: manejadores de XML usados a la hora de leer el fichero de monumentos de la junta en el caso de lugar y para leer/escribir la ruta en el caso de ruta. Se utiliza SAX para implementarlos[1].

## 6.4. Casos de Uso

A modo explicativo, en Android hay varias formas de proporcionar realimentación al usuario, en la aplicación se usan dos: Snackbar[7] y Toast[17]. Ambos muestran mensajes al usuario, aunque su comportamiento difiere un poco. A lo largo de la sección, cada vez que se menciona «Informar al usuario» se está refiriendo a mostrarle un Snackbar o un Toast con la información requerida (por ejemplo, «No estás conectado a Internet»). Se deja de esta forma indicado para no sobrecargar la explicación.

A continuación se describe el comportamiento de la aplicación al implementarse los distintos casos de uso.

Los diagramas de secuencia de acceso a datos que quedan indicados en esta sección se encuentran en el apéndice Diagramas de acceso a datos.

### 6.4.1. Iniciar Aplicación



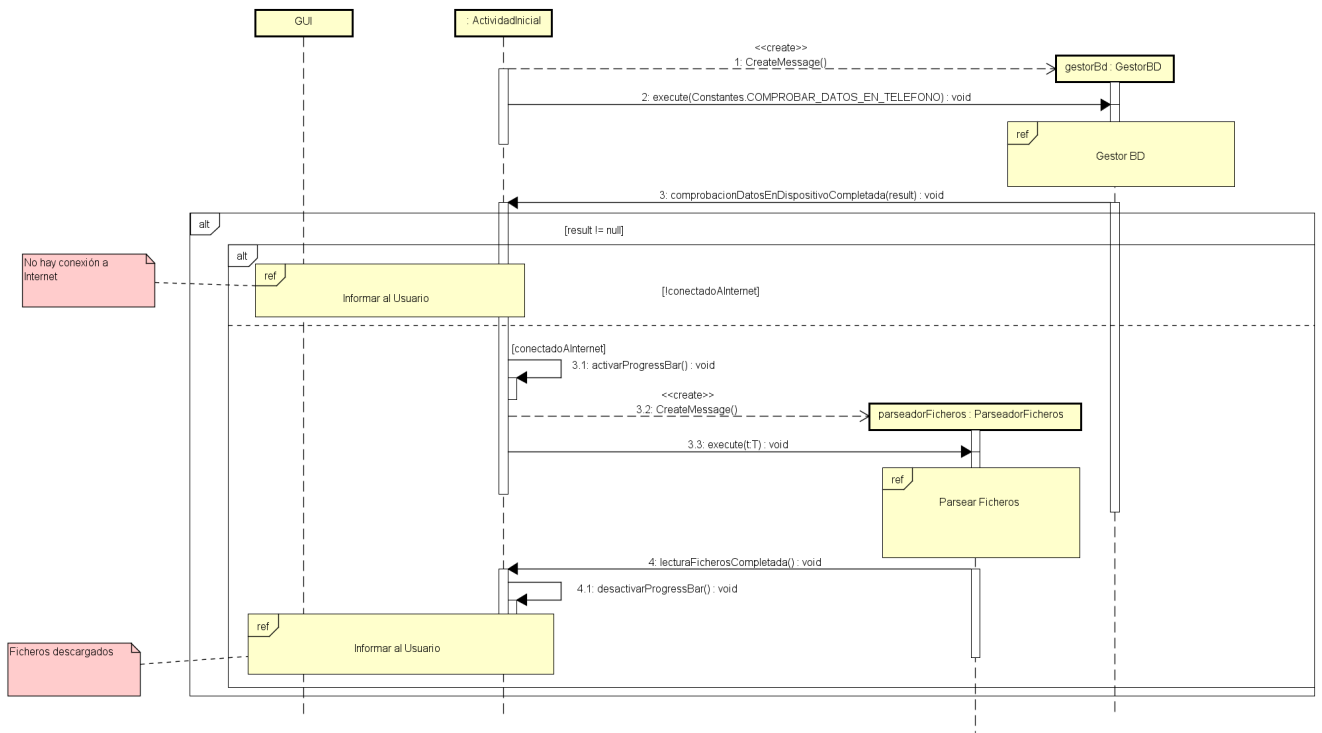


Figura 11: Iniciar Aplicación

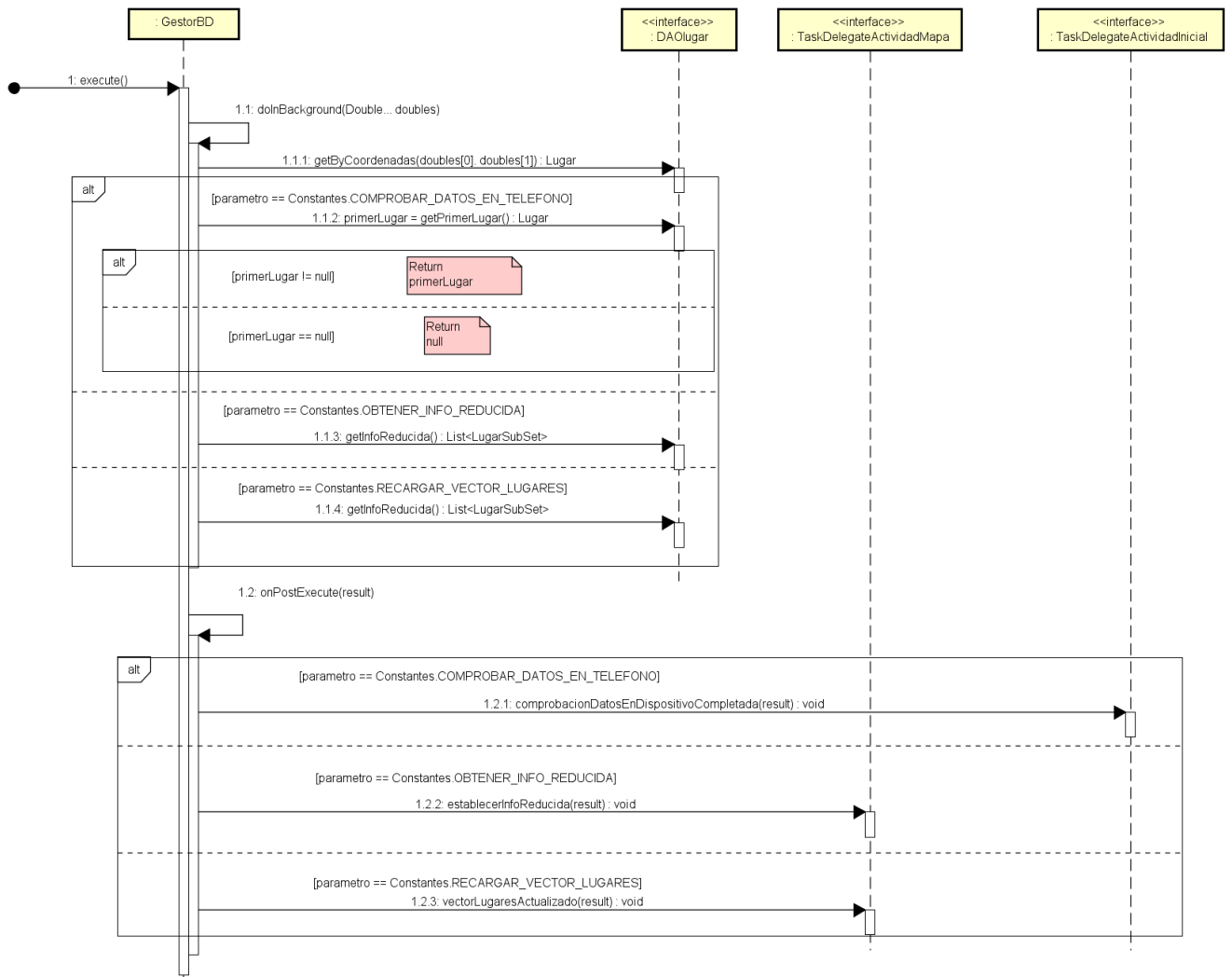


Figura 12: Gestor BD

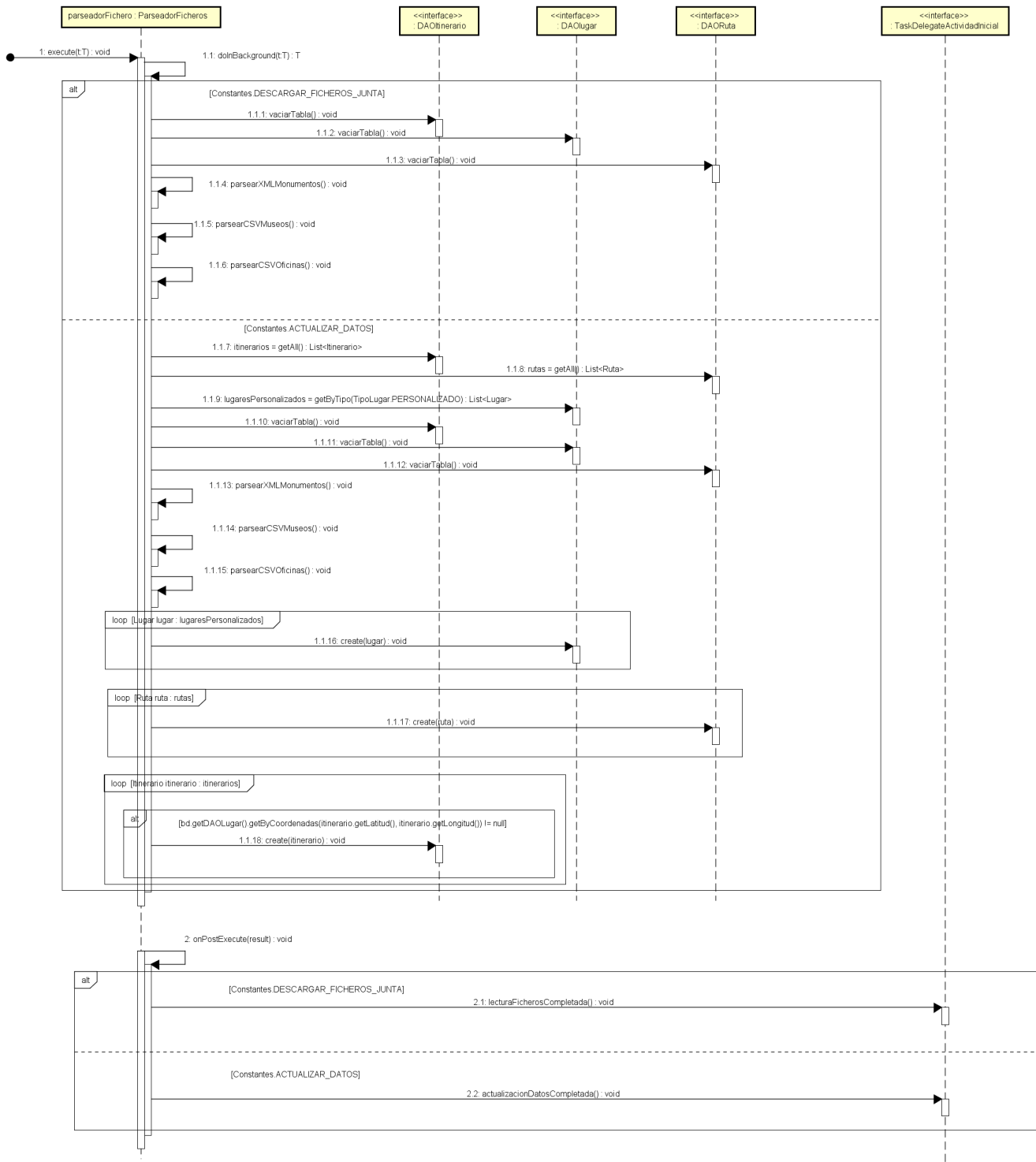
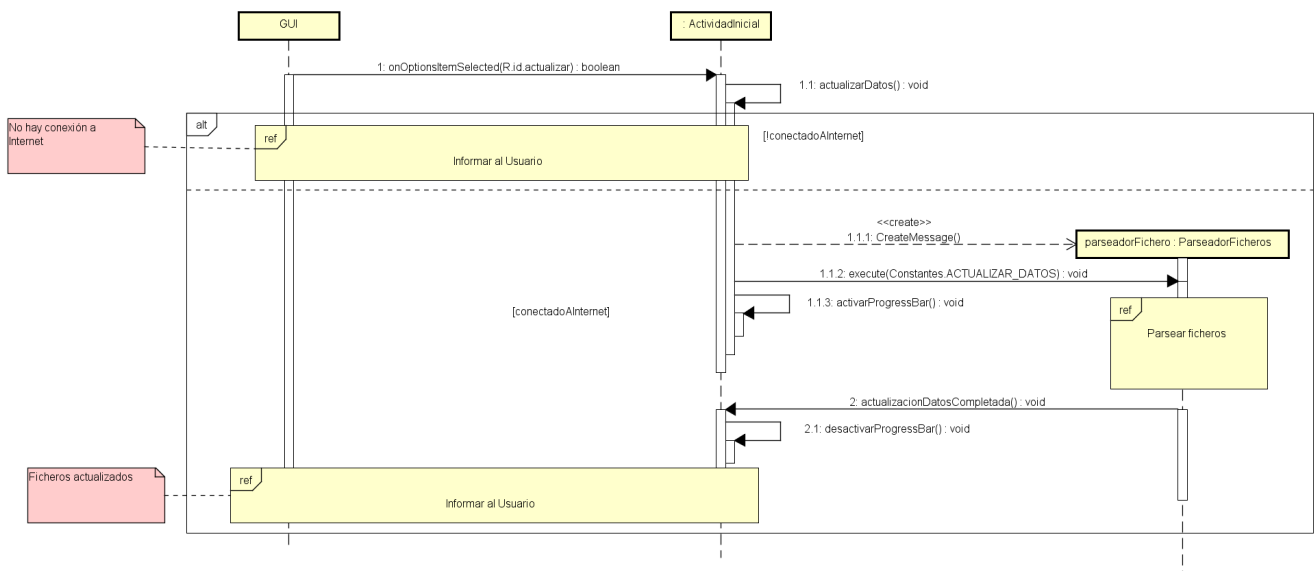


Figura 13: Parsear Ficheros

En este caso, al iniciarse, `ActividadInicial` comprueba si existen datos en el dispositivo [Gestor BD]. Si el primer lugar de la tabla `Lugar` es distinto de `null` se entiende que existen datos en el dispositivo, si no es porque es el primer inicio o porque se han eliminado los datos manualmente. Si existen datos no se hace nada, pero si no existen se comprueba la conexión a Internet. Si no se está conectado se informa al usuario, si se está conectado y el usuario confirma se incorporan los lugares que hay en los ficheros de la junta a la base de datos, [Parsear Ficheros], leyendo la información de remoto para cada uno de los archivos (Monumentos, Museos y Oficinas de Información Turística, en XML y CSV). Se activa un `ProgressBar` para impedir que el usuario intente abrir el mapa y se muestra realimentación sobre el progreso de la descarga de datos. Una vez finalizado, se desactiva el `ProgressBar` y se habilitan las demás opciones.

### 6.4.2. Actualizar Base de Datos



**Figura 14:** Actualizar base de datos

El usuario selecciona la opción «Actualizar» y se comprueba la conexión a Internet. Si no se está conectado se informa al usuario, si está conectado se activa el `ProgressBar` como en el caso de uso anterior y se actualiza la base de datos [Parsear Ficheros]; primero se guardan los itinerarios, las rutas y los lugares personalizados que existan en la base de datos, después se vacían las tres tablas para evitar problemas de inconsistencia, a continuación se parsean los ficheros de la junta y por último se vuelve a copiar la información de rutas, itinerarios y lugares personalizados, comprobando antes de agregar un itinerario si ese lugar sigue existiendo, para evitar errores de clave foránea. Una vez se ha actualizado, se desactiva el `ProgressBar` y se informa al usuario.

Tanto en este caso de uso como en el anterior, antes de realizar la acción, se muestra un mensaje al usuario y se solicita confirmación. Se decide hacer esto para aumentar la usabilidad de la aplicación.

### 6.4.3. Abrir Mapa

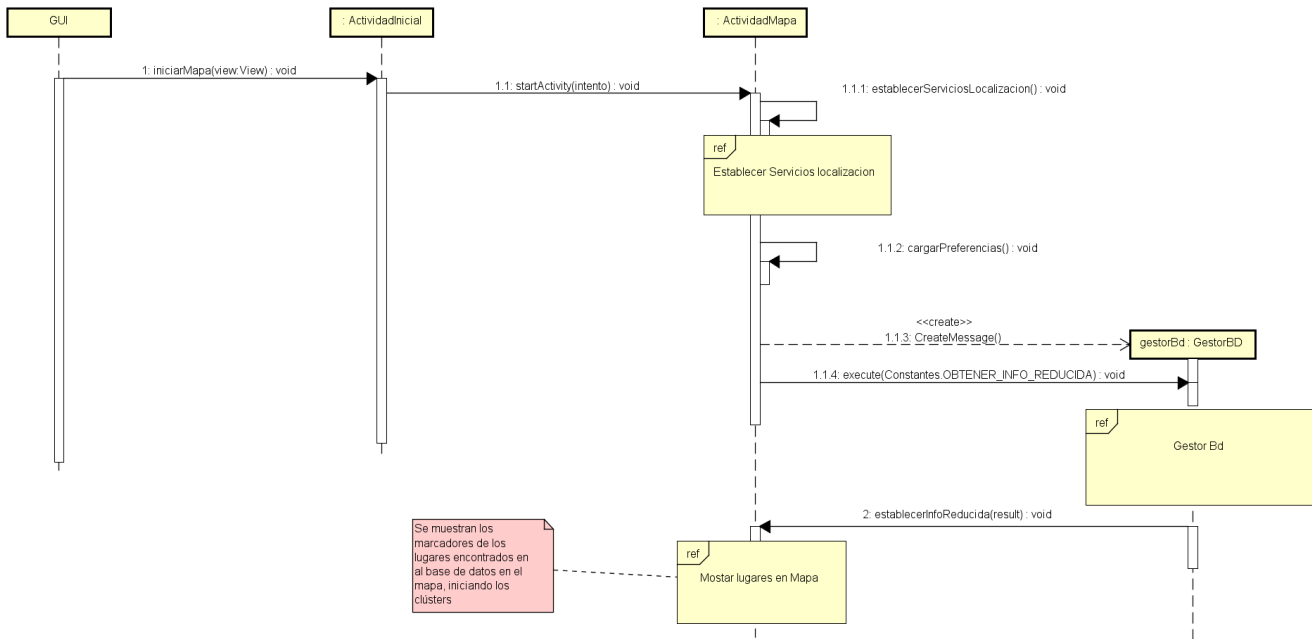


Figura 15: Abrir Mapa

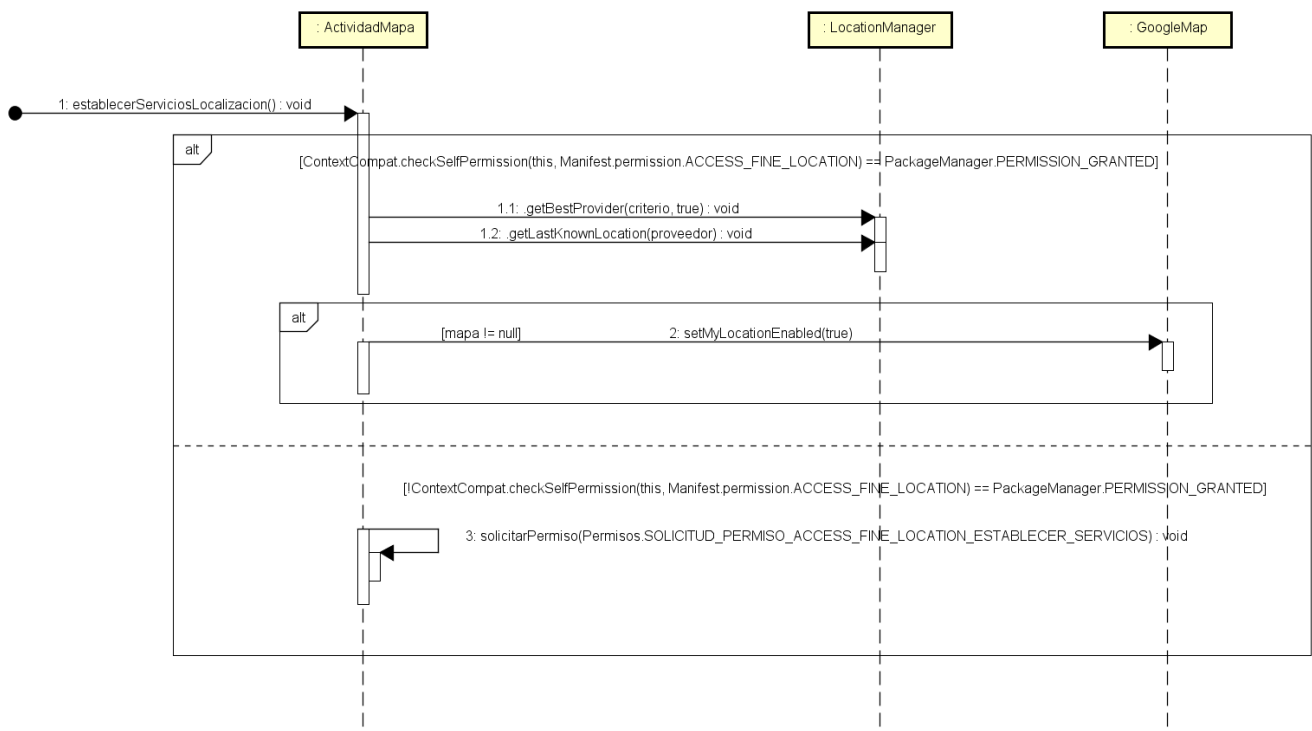


Figura 16: Establecer servicios de localización

El usuario selecciona la opción «Abrir Mapa», se inicia ActividadMapa, que intenta establecer los servicios de localización [Servicios de Localización]. Primero se comprueba si se tienen permisos de acceso a la ubicación del usuario, si no se tienen se solicitan. En caso de tenerlos, se busca el mejor proveedor en base a un criterio (gratis, sin altitud mínima, alta precisión) y se intenta obtener la última ubicación conocida, además de habilitar la capa de localización del mapa si no lo estaba. Una vez realizado se cargan en la aplicación las preferencias del usuario y se obtiene

de la base de datos los lugares que haya [Gestor BD], para por último mostrarlos en el mapa. El proceso de mostrado en el mapa involucra situar los marcadores en las coordenadas de los lugares en función de los valores de visibilidad definidos por el usuario en las preferencias, e iniciar los clusters, así como realizar varias acciones sobre el mapa (solicitar el mapa a Google Maps, esperar a que esté listo, establecer tipo, etc).

#### 6.4.4. Agregar Lugar

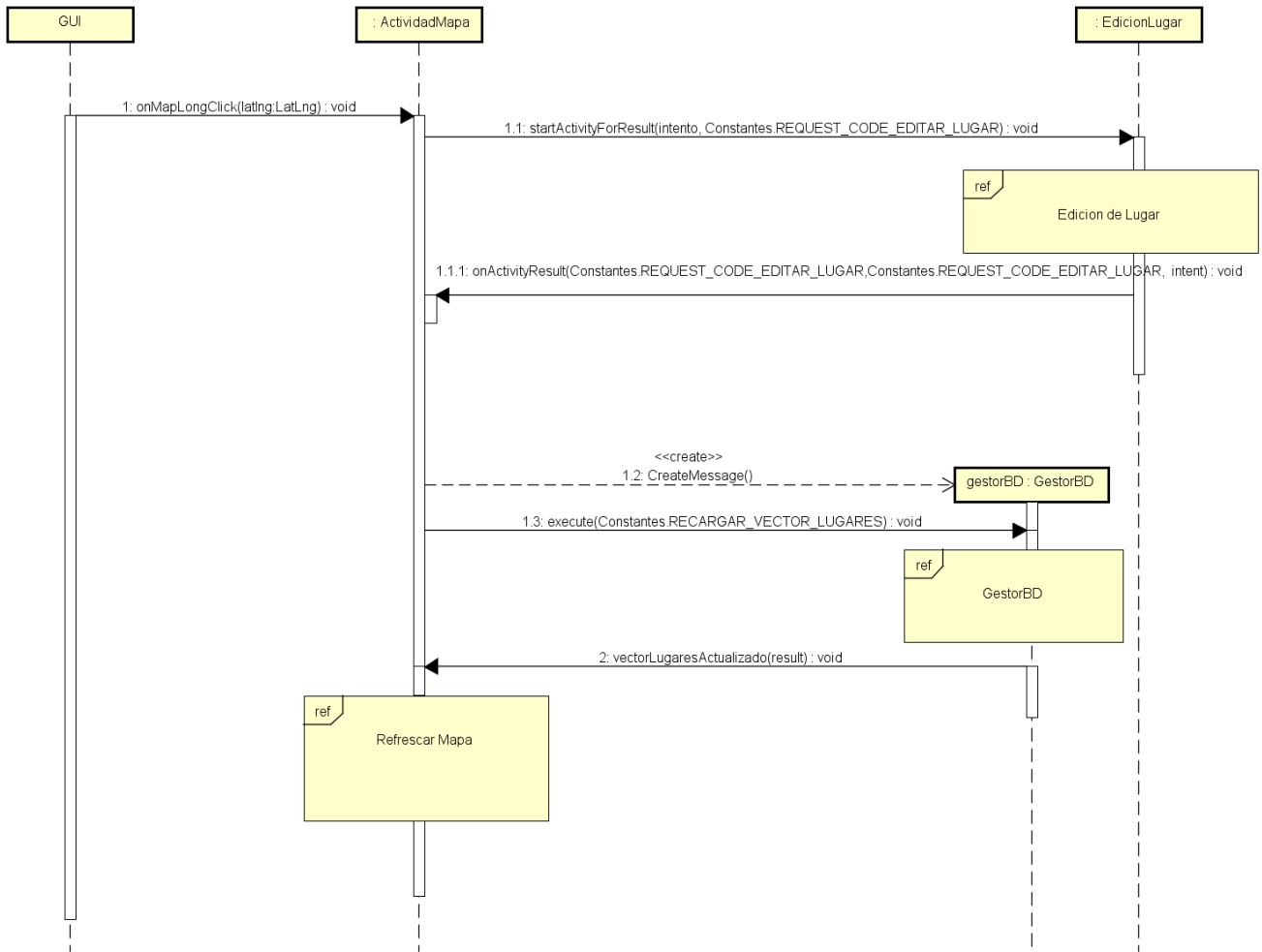
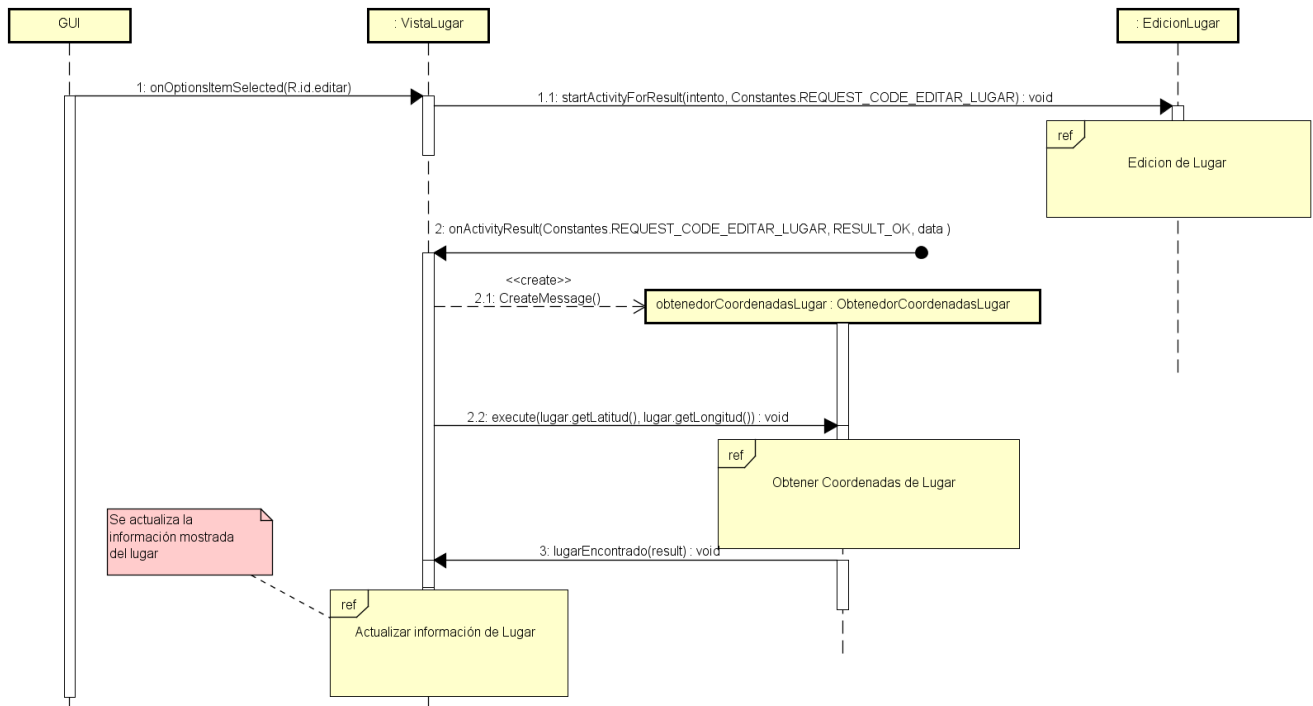


Figura 17: Agregar Lugar en mapa



calización de Google[21]. Se establecen los atributos del lugar en base a lo que haya introducido el usuario y se agrega el lugar a la base de datos. Una vez hecho, se vuelve a ActividadMapa, se recupera la información de los lugares de la base de datos y se refresca el mapa (eliminar marcadores, volver a ponerlos, re-inicializar el cluster, etc).

### 6.4.5. Editar Lugar



**Figura 19:** Editar Lugar

El usuario selecciona la opción editar lugar en el menú de VistaLugar (el lugar es de tipo «Personalizado», si no no se muestra la opción como disponible), y se abre EdicionLugar [Edición de Lugar]. Esta actividad busca el lugar en la base de datos y, al encontrarlo, establece los valores de los campos del lugar. El usuario puede pulsar confirmar o descartar los cambios, si los descarta la actividad finaliza. Si los confirma se comprueba que al menos haya introducido el nombre, si no lo ha introducido se le informa. Al estarse editando un lugar, la aplicación actualiza ese lugar con los nuevos valores y vuelve a VistaLugar, que actualiza la información mostrada con los nuevos datos introducidos.

## 6.4.6. Eliminar Lugar

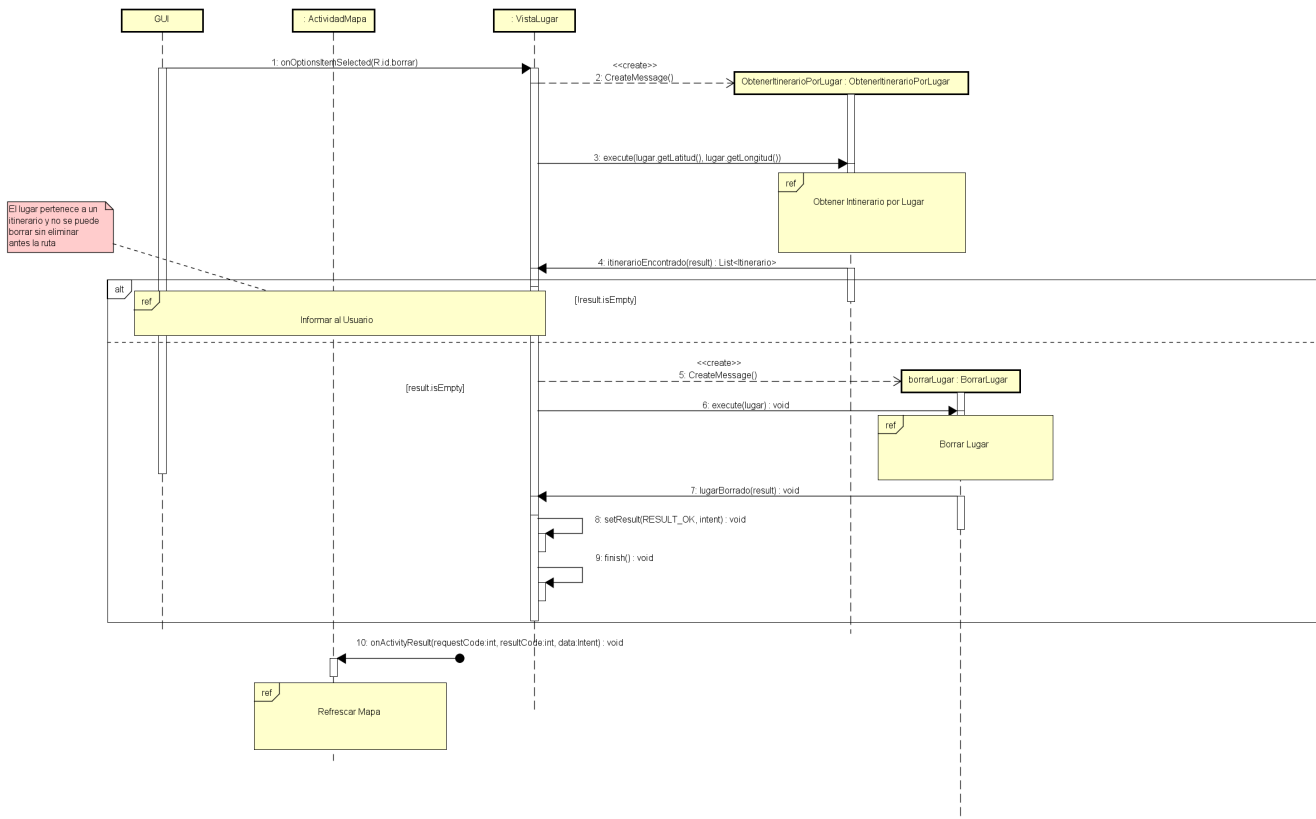


Figura 20: Eliminar Lugar

El usuario selecciona la opción eliminar lugar en el menú de VistaLugar (el lugar es de tipo «Personalizado», si no, no se muestra la opción como disponible). Se buscan los Itinerarios de ruta para ese lugar. Si está en alguno (pertenece al menos a una ruta), se informa al usuario y no se continúa con el borrado. Si no está agregado a ninguno, se borra el lugar de la base de datos, se vuelve a ActividadMapa y se refresca el mapa (eliminar marcadores, volver a ponerlos, re-inicializar el cluster, etc).



### 6.4.7. Trazar Ruta a Lugar

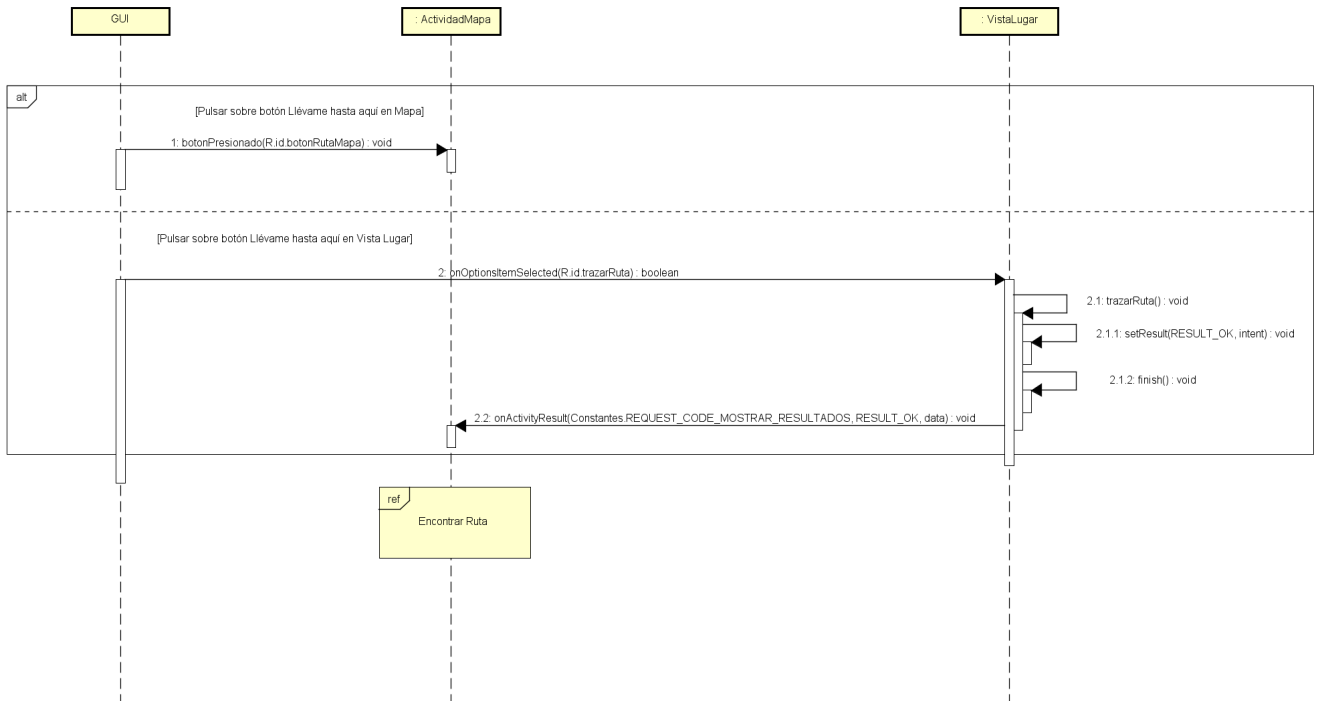


Figura 21: Trazar ruta a Lugar

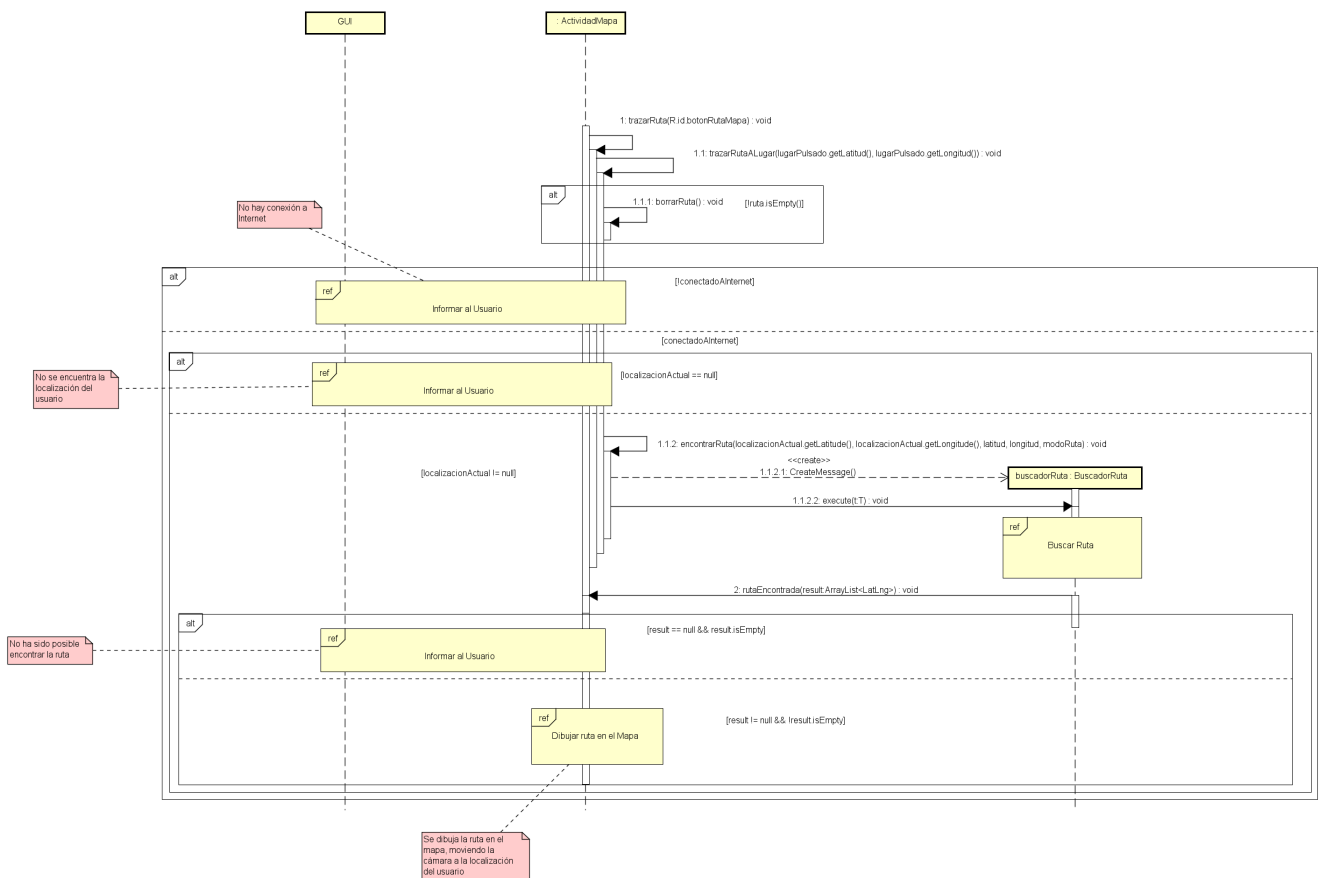


Figura 22: Encontrar Ruta



**Figura 23:** Buscar Ruta

El caso de uso se puede acceder desde dos lugares diferentes: desde ActividadMapa al pulsar sobre un marcador y pulsar el botón «Llévame hasta aquí» y desde VistaLugar, al pulsar sobre la opción de menú «Llévame hasta aquí». En ambos casos se vuelve a ActividadMapa y si hubiese alguna ruta ya dibujada, se borra. Si no se está conectado a Internet o no se tiene activada la localización (o no se ha dado permiso para usarla), se informa al usuario y no se busca la ruta. En los demás casos se accede a la API de rutas de Google y se intenta obtener la ruta entre la localización de origen y la de destino. Si no es posible encontrar la ruta se informa al usuario, en caso contrario, se dibuja en el mapa con los valores definidos por el usuario en las preferencias.

Existe un «fallo» en este método aunque no es culpa de la aplicación. La API de rutas de Google ofrece dos versiones, gratuita y de pago. La versión gratuita ofrece un número determinado de consultas al día y no permite a veces realizar peticiones muy seguidas. A pesar de que se ha intentado buscar el número exacto de peticiones realizables y el tiempo de *cooldown* no ha sido posible encontrarlo. Hay que decir que este «fallo» se da en pocas ocasiones y es externo a la aplicación, por lo que no se profundizó demasiado en implementar un «workaround» que lo solucionase.

A pesar de existir la opción de elegir transporte público como medio de transporte a la hora de calcular la ruta, se desestimó, ya que sólo se encuentra en recorridos entre grandes ciudades y alrededores, dejando la inmensa mayoría de lugares que incorpora la aplicación, desconectados de este servicio.

### 6.4.8. Mostrar Lugar en Mapa

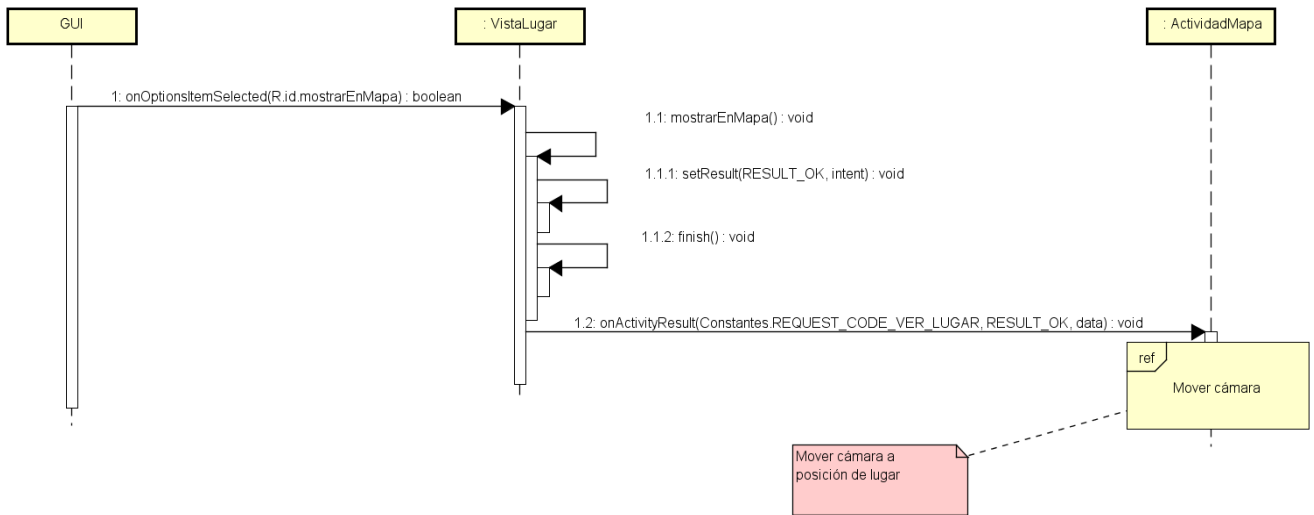


Figura 24: Mostrar Lugar en Mapa

El usuario selecciona la opción de menú «Mostrar en Mapa», se accede a ActividadMapa y se mueve la cámara a las coordenadas del lugar elegido.

### 6.4.9. Llamar número

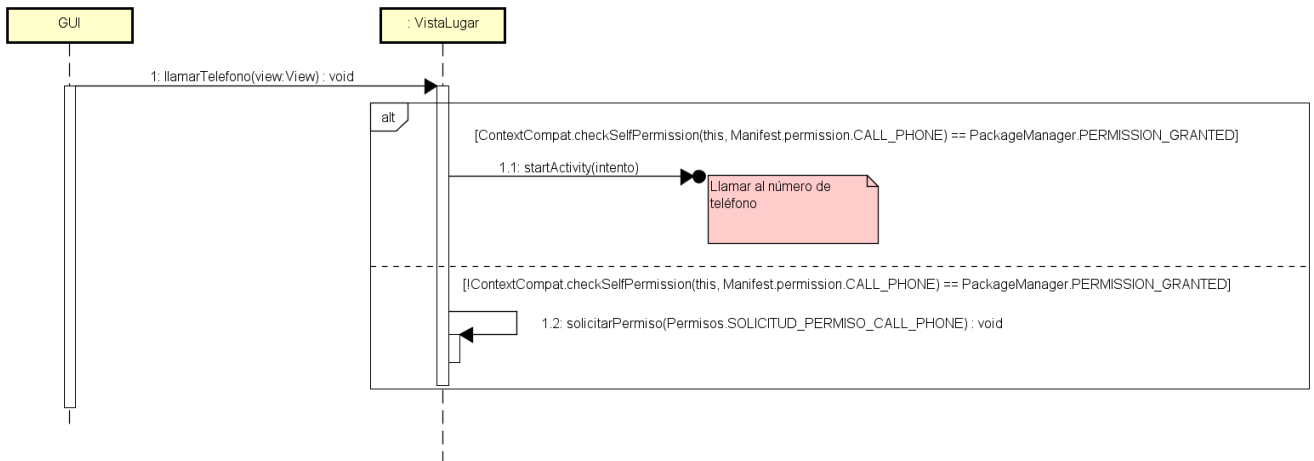


Figura 25: Llamar Número

El usuario pulsa sobre el número de teléfono de un lugar (si no tiene no aparece y no se puede pulsar), y se comprueban los permisos. Si no se han dado, se solicitan; si se tienen se llama al número de teléfono.

### 6.4.10. Abrir correo



Figura 26: Abrir Correo

El usuario pulsa sobre la dirección de correo de un lugar (si no tiene no aparece y no se puede pulsar), y se abre el gestor de correo elegido por él, creando un nuevo borrador con destinatario ese correo.

#### 6.4.11. Abrir web

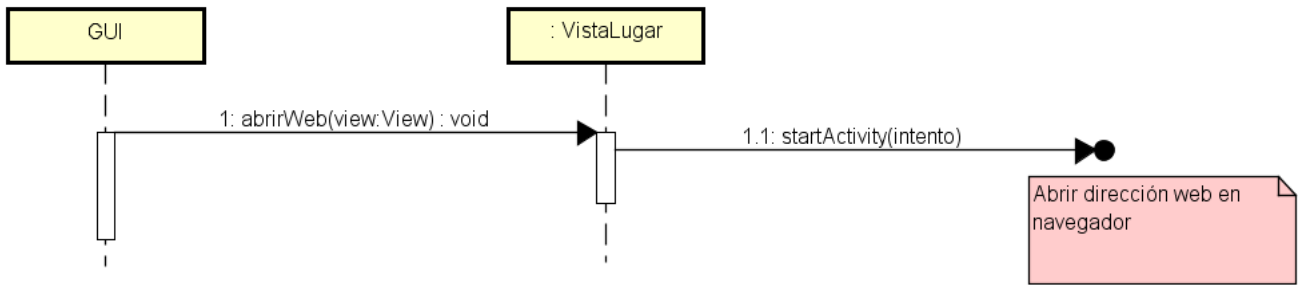


Figura 27: Abrir Web

El usuario pulsa sobre la dirección web de un lugar (si no tiene no aparece y no se puede pulsar), y se abre el navegador elegido por él, mostrando la página web.

#### 6.4.12. Mostrar Información de Lugar

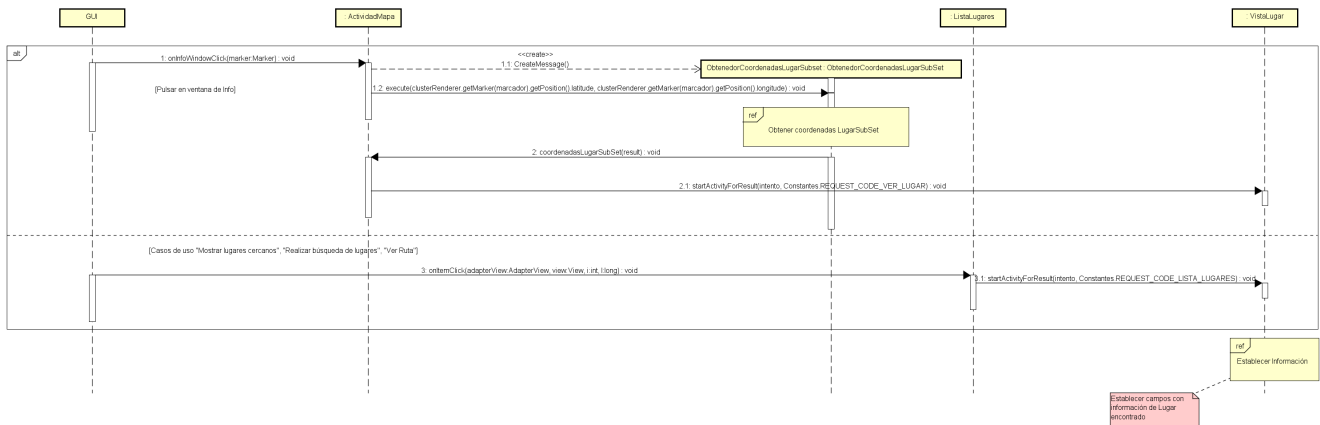
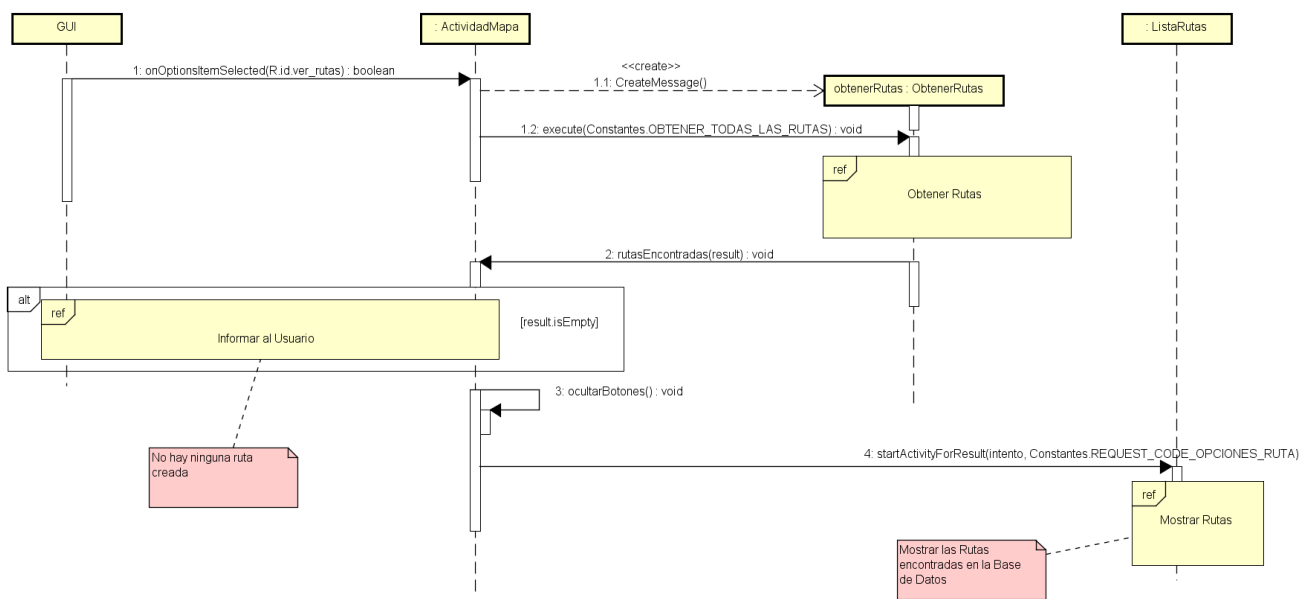


Figura 28: Mostrar Información de Lugar

El usuario puede acceder desde dos actividades diferentes: ActividadMapa, al pulsarse sobre la ventana de información de un marcador o desde ListaLugares al pulsar sobre uno de los elementos de la lista. Si es desde ActividadMapa se buscan primero las coordenadas del lugar, ya que no se pueden obtener directamente debido a los clusters. Una vez hecho, se inicia VistaLugar y se completan los campos con la información de lugar. Si no se encuentra un campo se completa con «No disponible» (este valor cambia dependiendo del idioma). Si se encuentra el número de teléfono, el email y/o la web, se muestran como enlaces para que el usuario pueda reconocerlos más fácilmente.

### 6.4.13. Mostrar Rutas



**Figura 29:** Mostrar Rutas

En ActividadMapa, el usuario selecciona la opción «Ver Rutas», la aplicación obtiene todas las rutas que se encuentran en la base de datos. Si no se han encontrado rutas, se informa al usuario. Se inicia ListaRutas y se muestran las rutas como elementos de una ListView (si no hay ninguna aparece vacío). Se decide abrir esta actividad aunque muestre una lista vacía ya que es la que permite agregar e importar rutas.

### 6.4.14. Ver Ruta

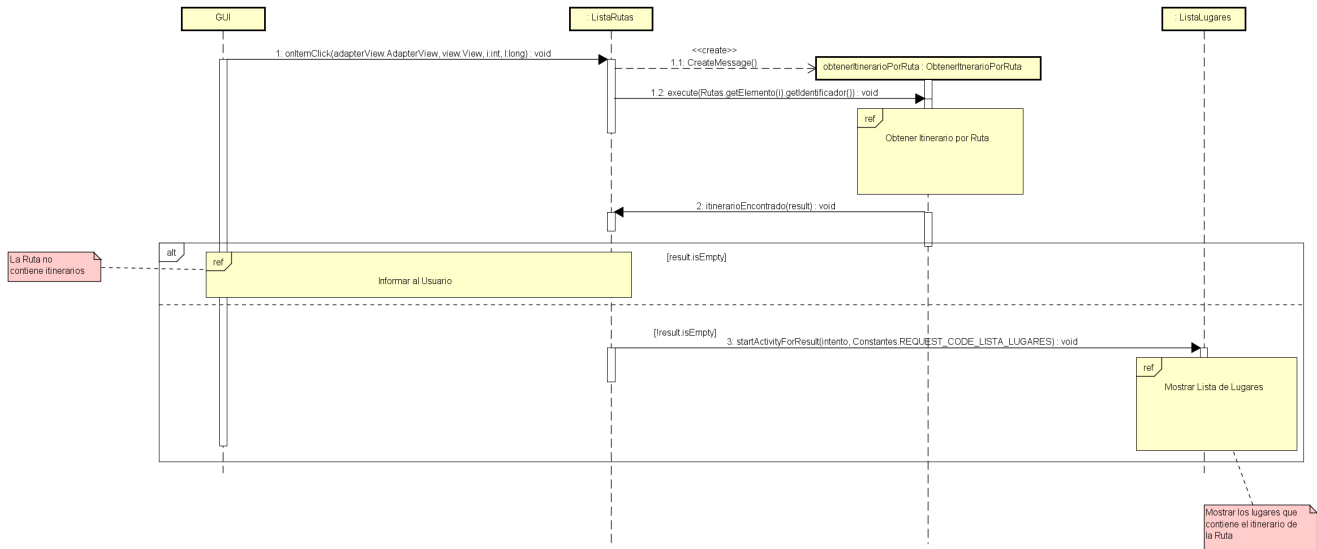


Figura 30: Ver Ruta

El usuario pulsa sobre uno de los elementos (una ruta) de la ListView de ListaRutas, se busca el itinerario para esa ruta en la base de datos. Si no tiene, se informa al usuario y se permanece en ListaRutas. En caso contrario, se muestran en ListaLugares los lugares que contenga esa ruta.

### 6.4.15. Agregar Ruta

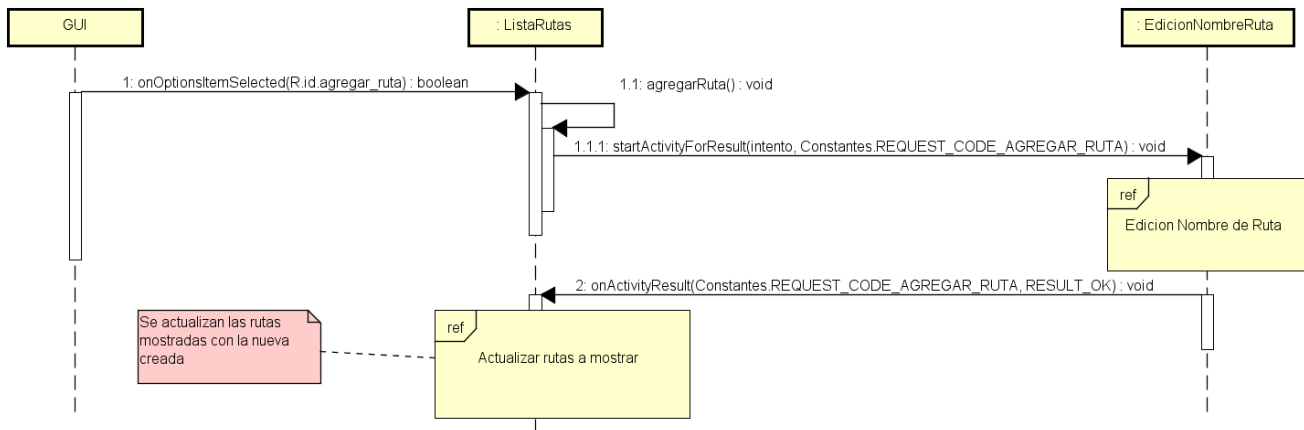
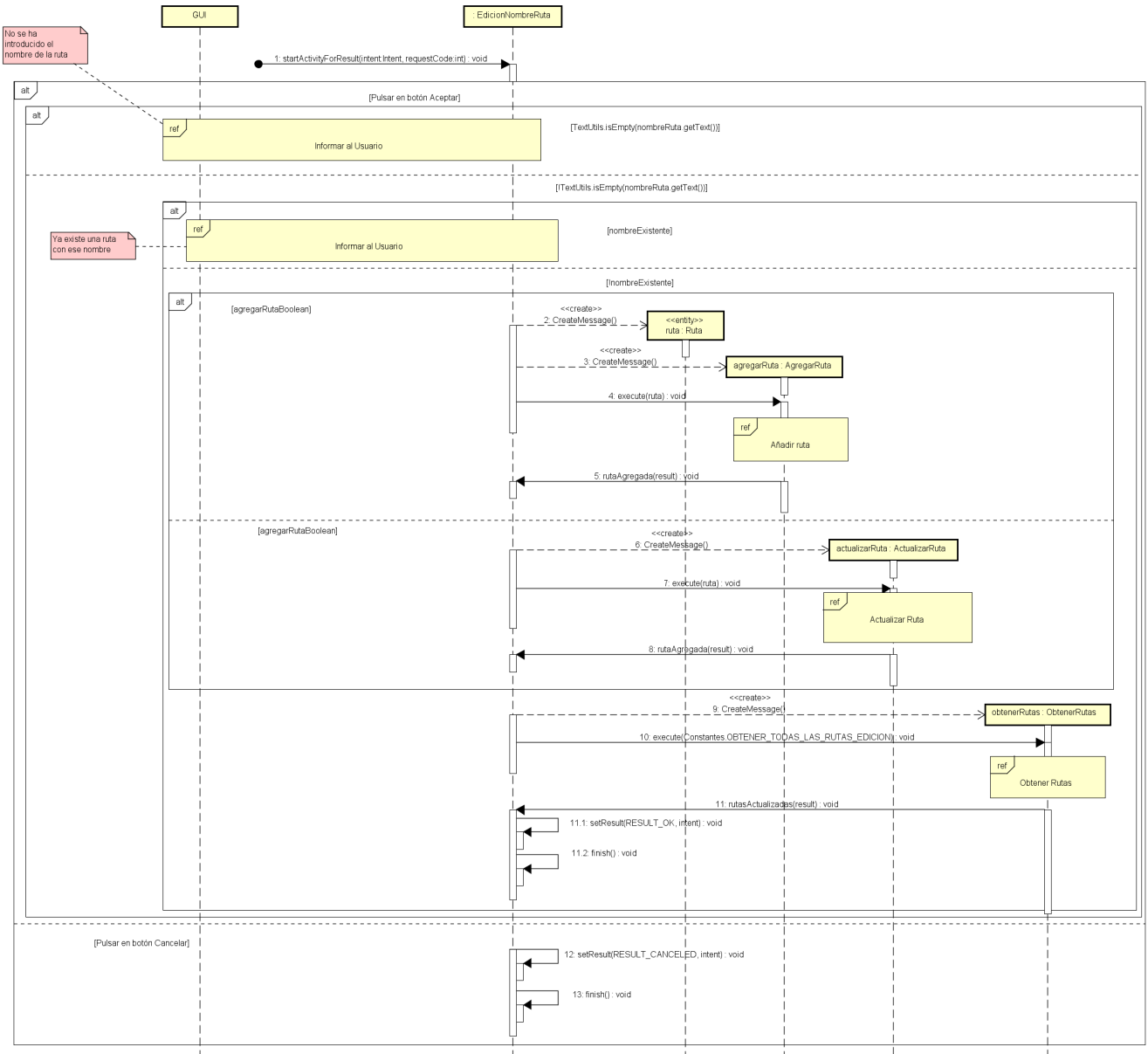


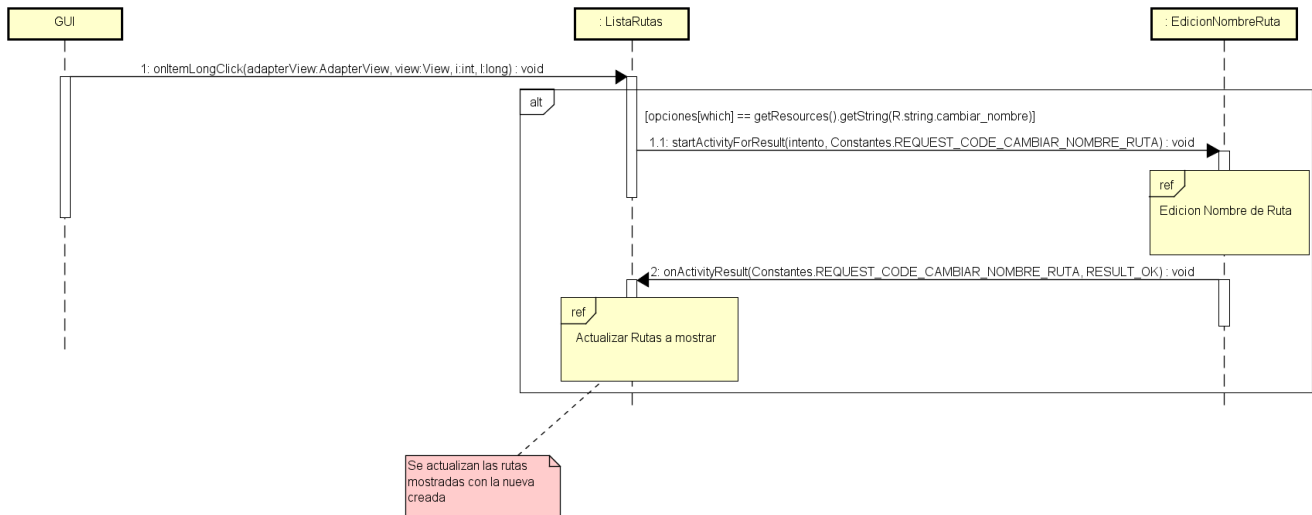
Figura 31: Agregar Ruta



**Figura 32:** Edición de Nombre de Ruta

El usuario selecciona la opción de menú de ListaRutas «Agregar Ruta», se inicia EdicionNombreRuta. Esta actividad comprueba si se está editando o agregando una ruta (si existe en la base de datos o no). Como se está agregando, el campo de nombre está vacío. El usuario puede confirmar o descartar los cambios, si los descarta la actividad finaliza y vuelve a ListaRutas. Si confirma, se comprueba que haya introducido el nombre y que este no pertenezca a ninguna ruta existente, se informa al usuario en estas situaciones. En caso contrario, se agrega la ruta a la base de datos y se vuelve a ListaRutas, actualizando las rutas mostradas por esta actividad.

### 6.4.16. Editar Nombre de Ruta

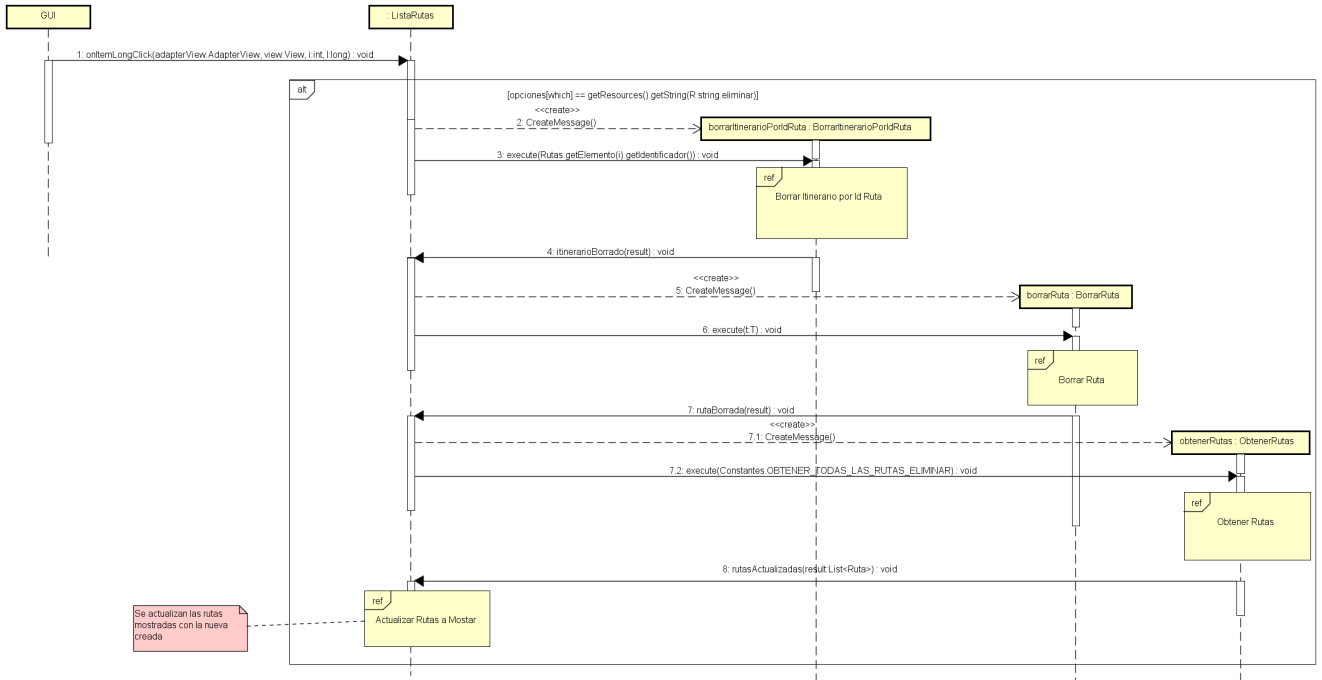


**Figura 33:** Editar Nombre de Ruta

El usuario mantiene pulsado sobre un elemento de la ListView de ListaRutas (una Ruta) y elige la opción «Editar nombre», se inicia EdicionNombreRuta [Edicion Nombre de Ruta]. Esta actividad comprueba si se está editando o agregando una ruta (si existe en la base de datos o no). Como se está editando, se completa el campo del nombre con el nombre de la ruta. El usuario puede confirmar o descartar los cambios, si los descarta la actividad finaliza y vuelve a ListaRutas. Si confirma, se comprueba que el nombre no pertenezca a ninguna ruta existente, se informa al usuario en esta situación. En caso contrario, se actualiza el nombre de la ruta en la base de datos y se vuelve a ListaRutas, actualizando las rutas mostradas por esta actividad.



### 6.4.17. Eliminar Ruta



**Figura 34:** Eliminar Ruta

El usuario mantiene pulsado sobre un elemento de la ListView de ListaRutas (una Ruta) y elige la opción «Eliminar Ruta», se solicita confirmación al usuario, si no confirma se vuelve a ListaRutas. Si confirma, se borra el itinerario de esa ruta de la base de datos, cuando termina se borra la ruta, cuando esta se ha borrado se actualizan las rutas mostradas por ListaRutas.



## 6.4.19. Editar Ruta

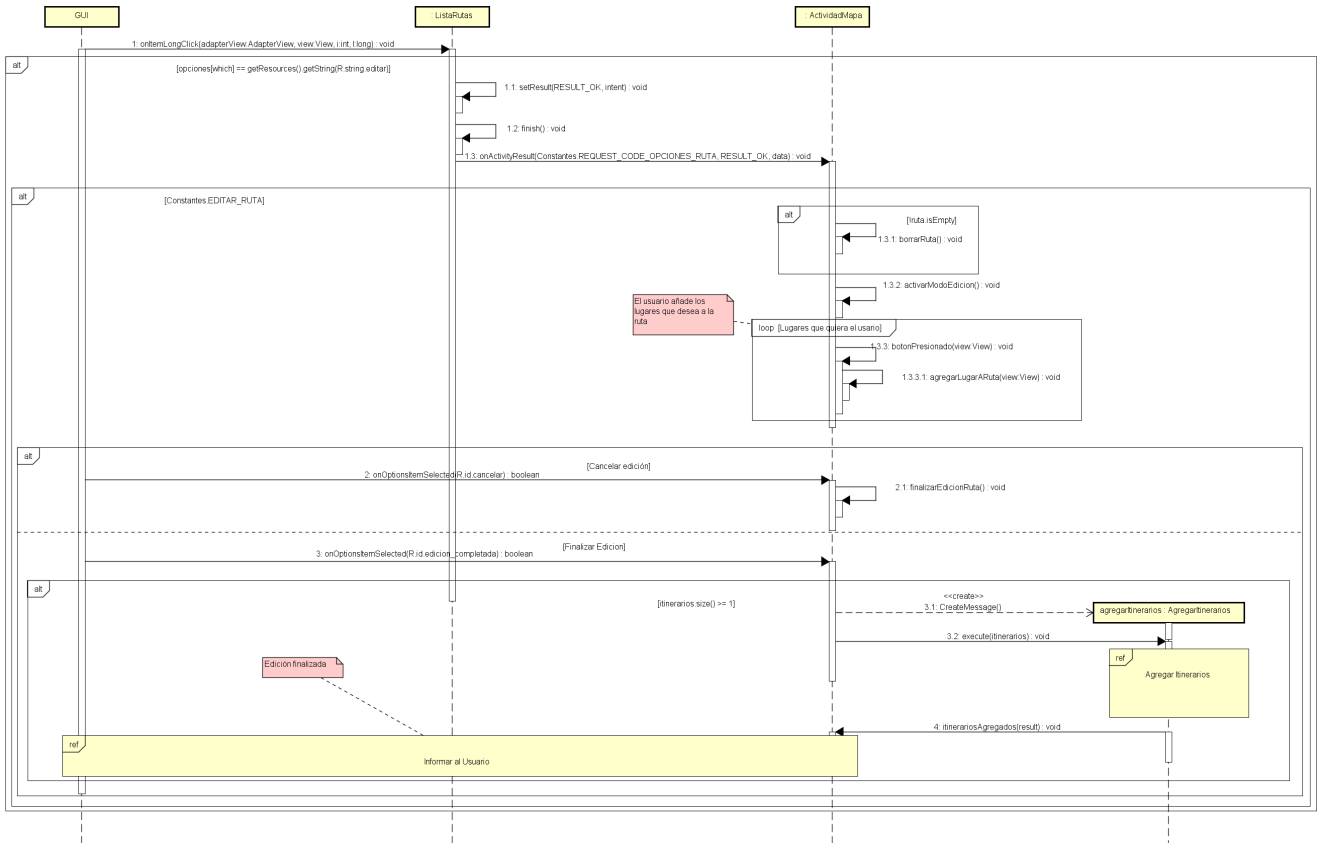


Figura 36: Editar Ruta

El usuario mantiene pulsado sobre un elemento de la ListView de ListaRutas (una Ruta) y elige la opción «Editar Ruta», se vuelve a ActividadMapa. Si hubiera dibujada alguna ruta se borra y se activa el modo de edición (deshabilitar opciones de trazar ruta, habilitar opciones de agregar a ruta y cambiar color de la aplicación). El usuario añade los lugares que desee a la ruta, pulsando sobre el marcador y sobre la opción «Agregar a Ruta» o, en VistaLugar, en la misma opción de menú. El usuario puede confirmar o descartar los cambios, si se descartan, se finaliza el modo de edición y no se modifica la ruta. Si se confirman y hay al menos un itinerario agregado, se agregan los itinerarios a la base de datos, se desactiva el modo de edición y se informa al usuario de que ha finalizado la edición.

### 6.4.20. Compartir Ruta

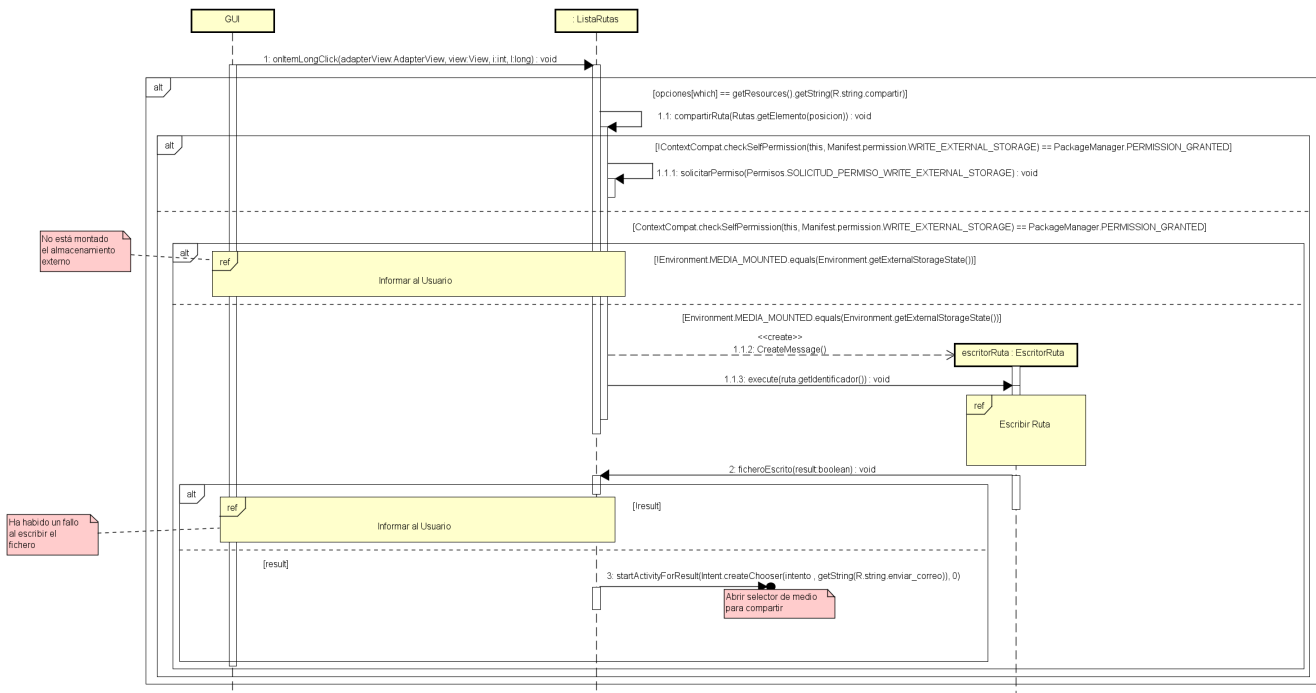


Figura 37: Compartir Ruta

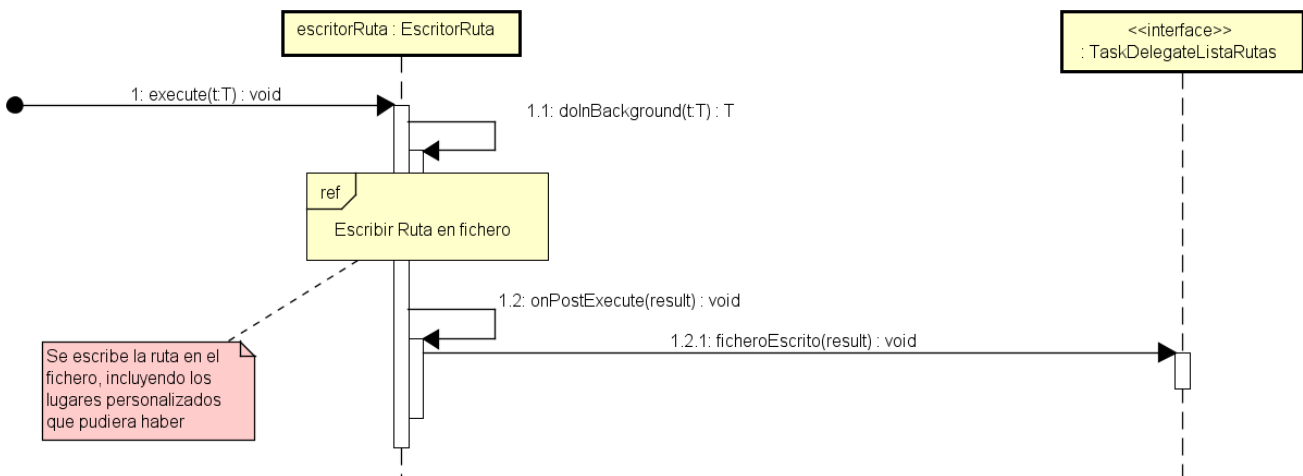


Figura 38: Escribir Ruta

El usuario mantiene pulsado sobre un elemento de la ListView de ListaRutas (una Ruta) y elige la opción «Compartir Ruta», se comprueban los permisos de escritura en almacenamiento externo, si no se tienen, se solicitan. Si se tienen se comprueba si está montado el almacenamiento externo, si no está montado se informa al usuario y se finaliza. Si está montado se escribe la ruta en un fichero, incluyendo su nombre, itinerario y lugares personalizados si los hubiera. Al finalizar la escritura, si han ocurrido errores se informa al usuario y no se comparte. Si no han ocurrido errores, se abre un selector de medio de compartición, el usuario elige y se añade como adjunto el fichero.

## 6.4.21. Importar Ruta

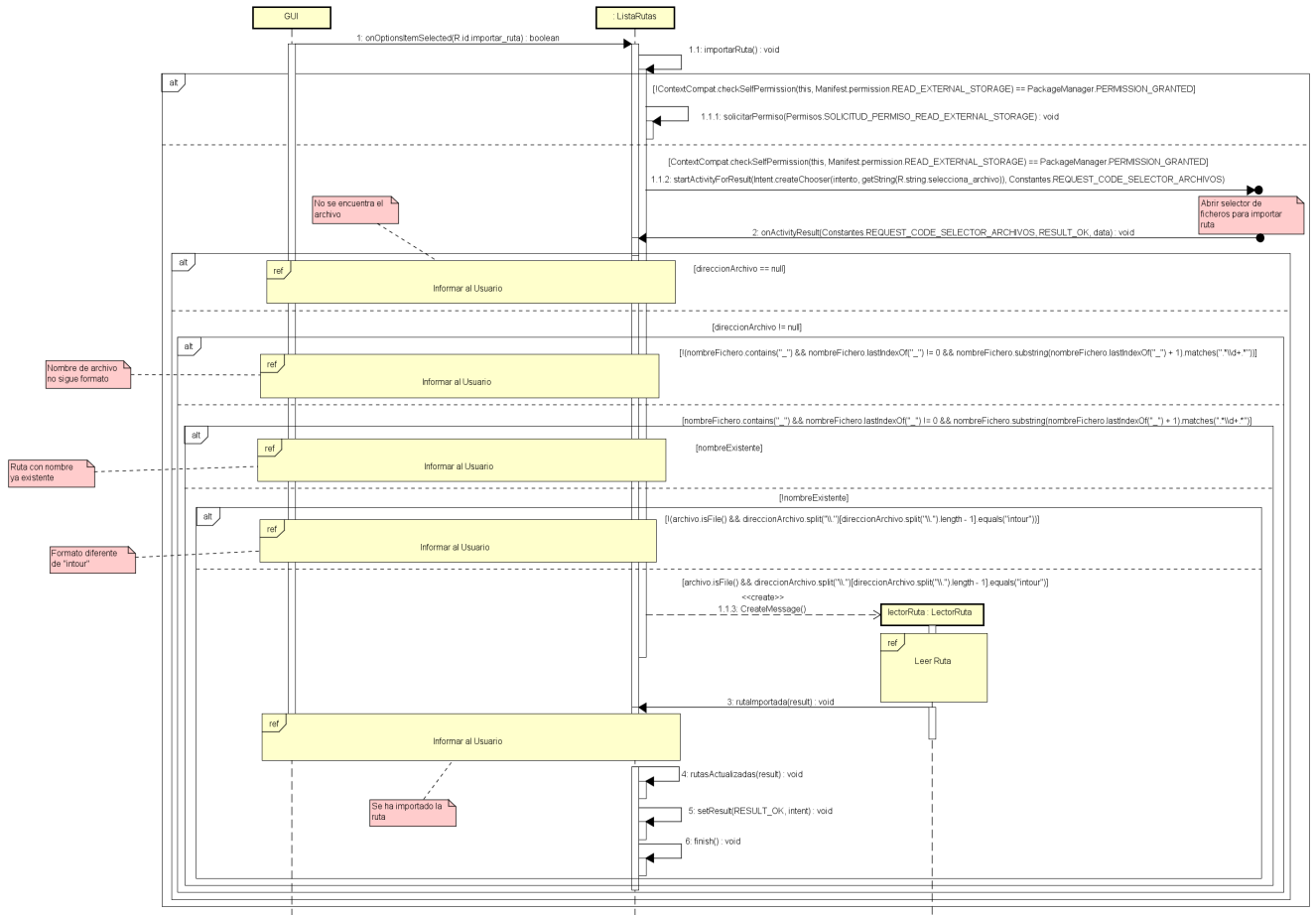


Figura 39: Importar Ruta

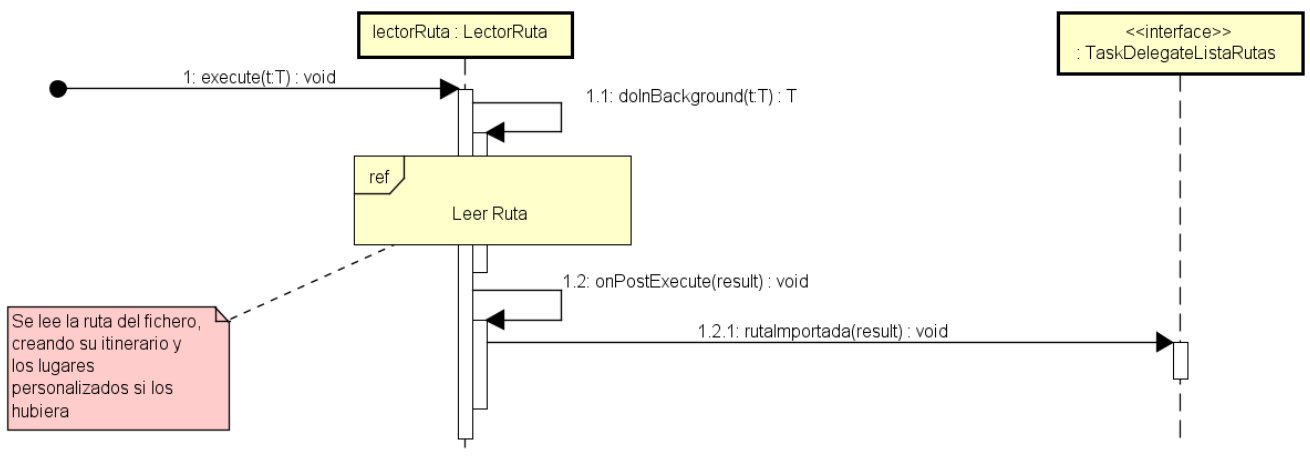


Figura 40: Leer Ruta

El usuario selecciona la opción de menú de ListaRutas «Importar Ruta», se comprueba si se tienen permisos de lectura de almacenamiento externo, si no se tienen se solicitan. En caso de tenerlos se deben dar unas condiciones para poder importar el fichero:

- La dirección de archivo no debe ser nula.

- El formato del nombre del fichero debe ser NombreDeRuta\_IdRuta
- El separador sólo puede ser «\_».
- El archivo elegido sólo debe ser archivo, no directorio.
- El formato del archivo debe ser «.intour».

Si no se da alguna de estas condiciones, se informa al usuario y no se lee el fichero. En caso de que se den todas las condiciones, se lee el archivo, agregando a la base de datos la ruta, su itinerario y los lugares personalizados en caso de que los hubiera. Una vez importada, se informa al usuario, se actualizan las rutas a mostrar y se refresca el mapa por si se han añadido nuevos lugares personalizados.

Tanto en este caso de uso como en el anterior, se manipulan ficheros de tipo «.intour», se explica su estructura en la sección [Formato de archivo .intour].

#### 6.4.22. Mostrar Lugares cercanos

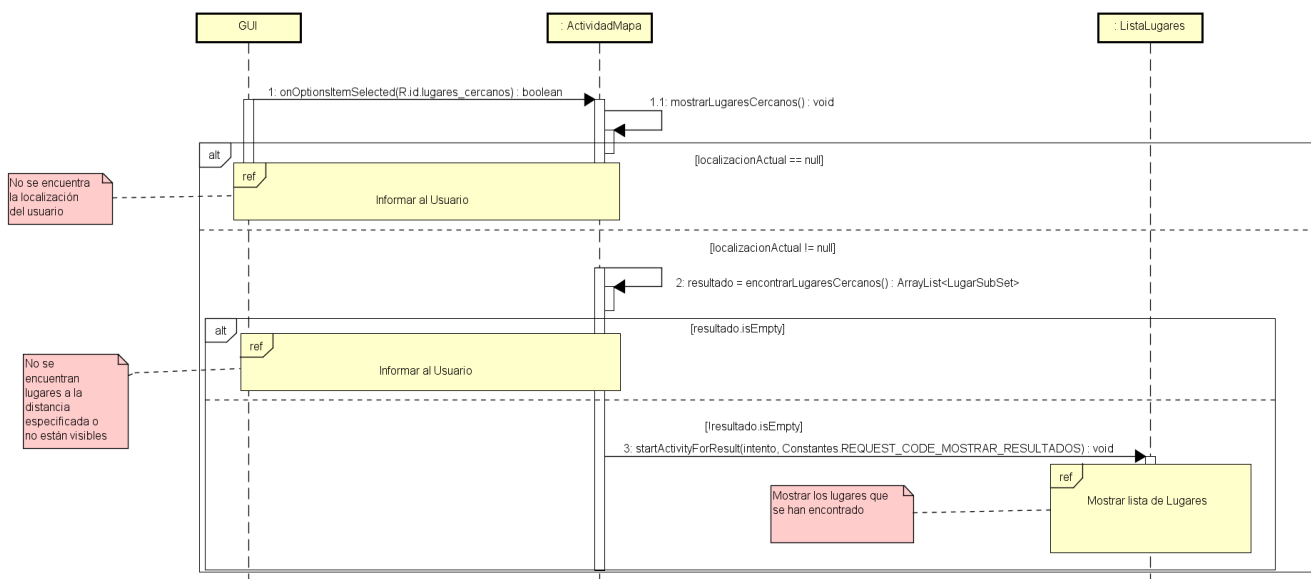


Figura 41: Mostrar Lugares cercanos

El usuario selecciona la opción de menú de ActividadMapa «Mostrar Lugares cercanos», se comprueba si el usuario tiene activada la localización, si no está se informa al usuario y no se continúa. Si está activada se buscan los lugares que están a igual o menor distancia que la distancia definida por el usuario en las preferencias. Una vez que se han encontrado se inicia ListaLugares y se muestran los lugares encontrados, en función de las preferencias de visibilidad del usuario, si no se han encontrado no se inicia y se informa al usuario.

### 6.4.23. Realizar búsqueda de lugares

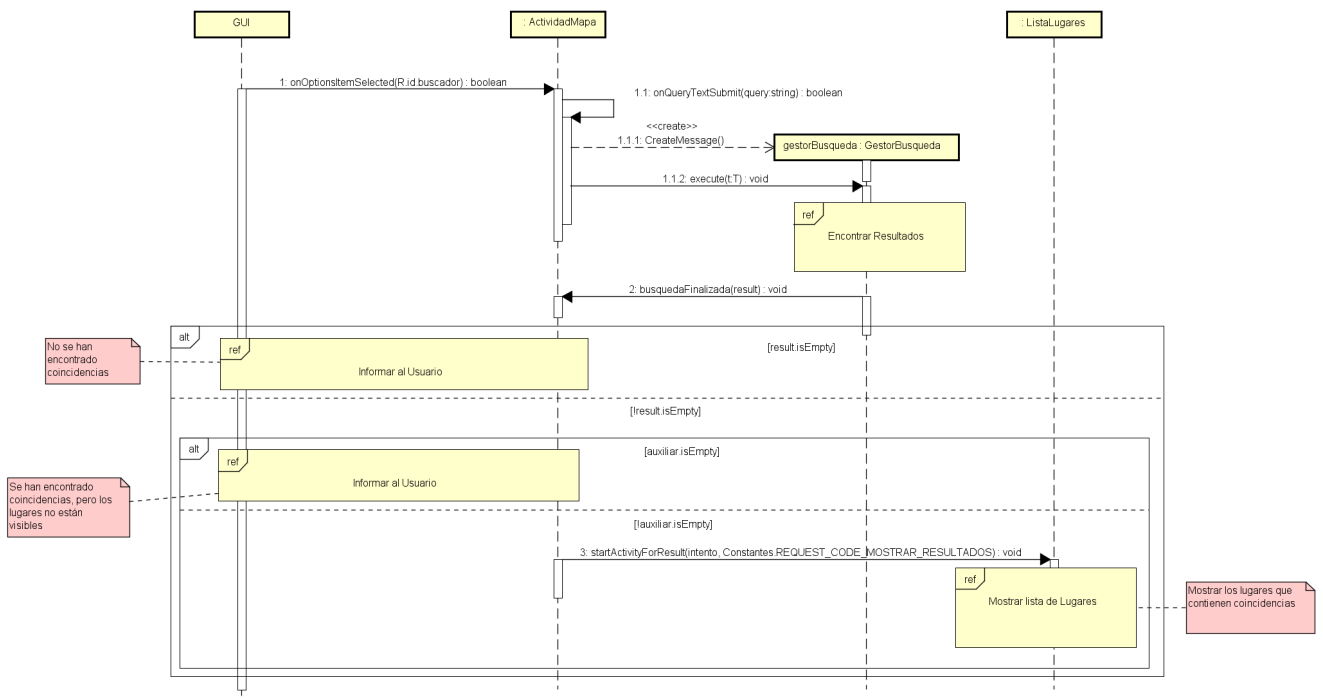


Figura 42: Realizar búsqueda de lugares

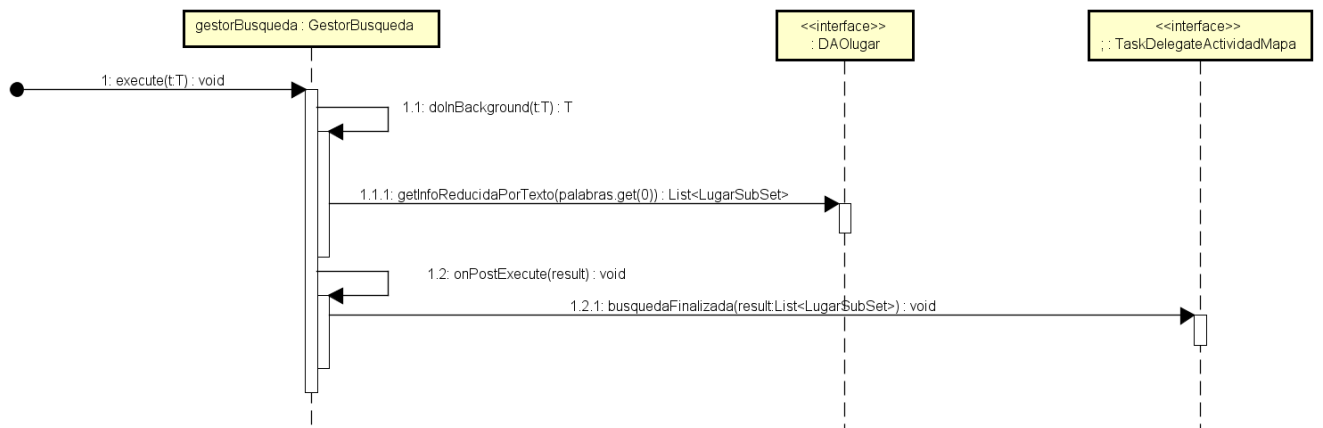
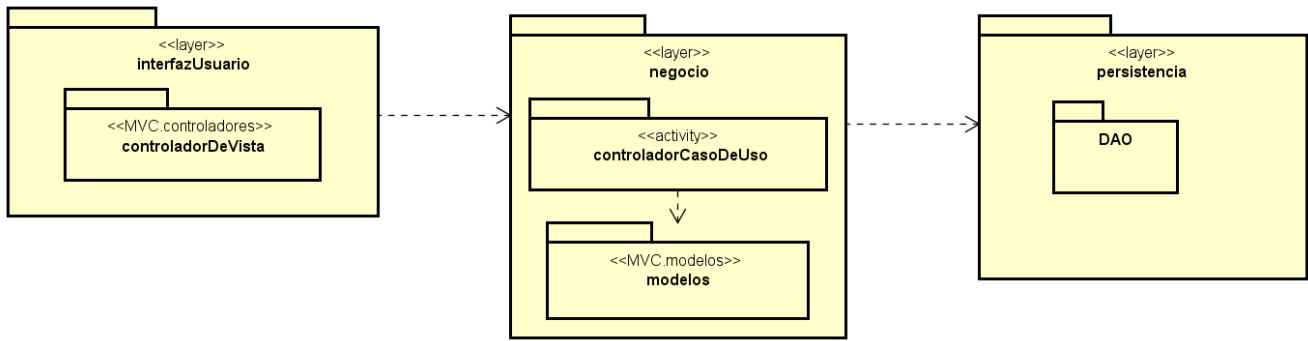


Figura 43: Encontrar resultados

El usuario selecciona el buscador, introduce una o varias palabras para buscar y selecciona la opción «Buscar». Se buscan coincidencias en los campos de Lugar Nombre, Localidad, TipoLugar y Descripción. El buscador elimina de la búsqueda las palabras repetidas, y para devolver un resultado deben aparecer en alguno de los campos todas las palabras introducidas (Por ejemplo, se busca «Iglesia de Valladolid», y las tres palabras deben aparecer entre Nombre, Localidad, TipoLugar o Descripción). Si no se han encontrado resultados, se informa al usuario, si se encuentran se eliminan aquellos que no resultan visibles debido a las preferencias de usuario. Si no queda ninguno, se informa al usuario, si no se inicia ListaLugares y se muestran los lugares encontrados.

## 6.5. Arquitectura y Despliegue

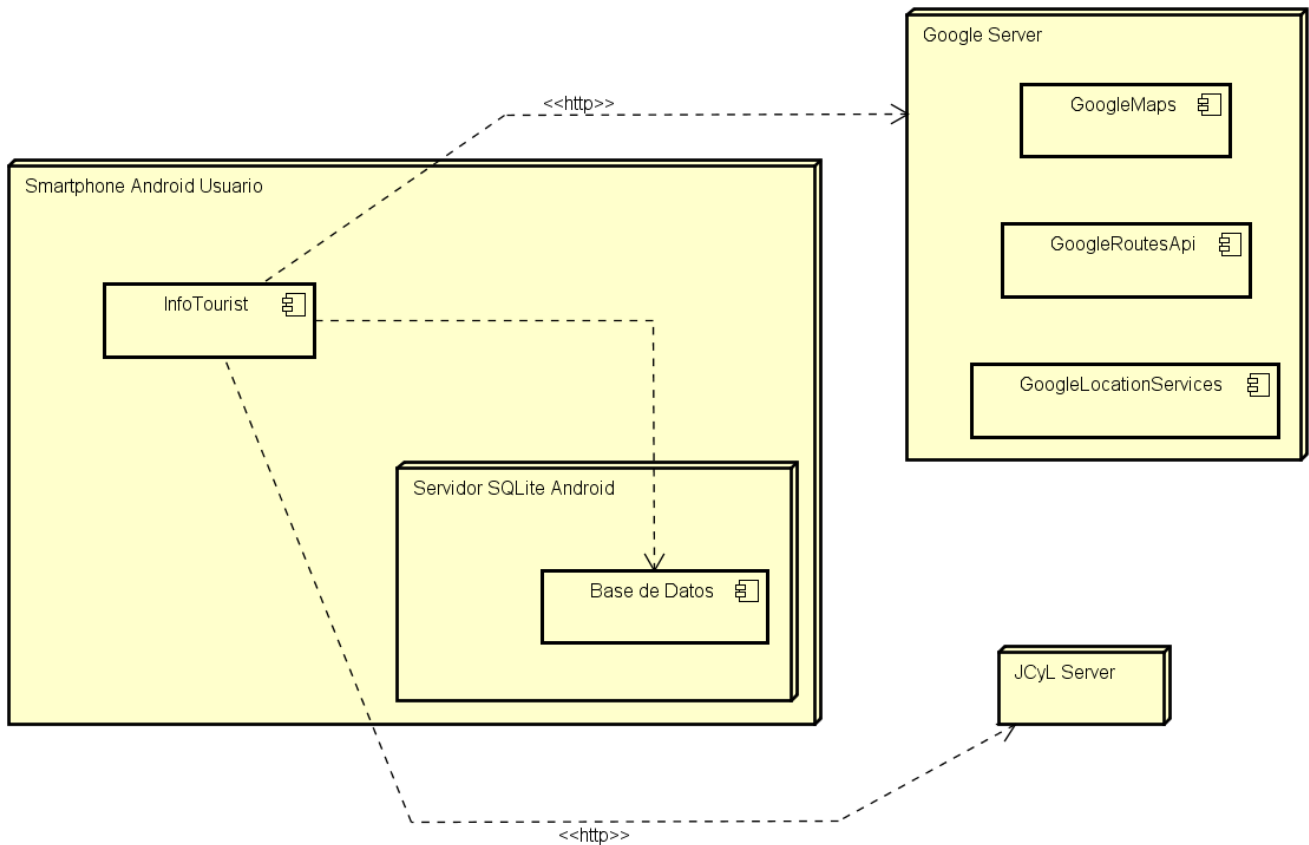


**Figura 44:** Diagrama de Arquitectura

S decide utilizar el patrón MVC (Modelo Vista Controlador), adaptado a las particularidades de Android[20] (uso de Layouts y Activities). El esquema del modelo es el siguiente:

- Capa de Interfaz de Usuario: se encarga de todo lo relacionado con la interfaz de usuario, presentar las vistas, actualizarlas, etc.
- Capa de Negocio: encargada de todos los aspectos del negocio de la aplicación, aquí se ubican los controladores de caso de uso, las clases del dominio, los modelos, etc.
- Capa de Persistencia: capa que tiene como cometido gestionar el acceso a datos y realizar las operaciones de inserción, actualización, borrado y recuperación de las distintas entidades que se encuentren en la base de datos.





**Figura 45:** Diagrama de Despliegue

El despliegue de la aplicación es bastante sencillo al encontrarse toda la aplicación en local, en el dispositivo del usuario. La aplicación se conecta con el servidor SQLite de Android para almacenar los datos, con el servidor de la Junta de Castilla y León para descargar los ficheros y con varios servidores de Google para obtener diferente información (el mapa, la ubicación y la ruta entre dos lugares).

## 6.6. Formato de archivo .intour

El archivo con la ruta compartida incluye el nombre de la ruta, su itinerario y los lugares personalizados que pudiese haber en la ruta. Se decide hacer esto para permitir a los usuarios compartir rutas con sus propios lugares. Se establece un formato personalizado llamado «intour» a pesar de que el contenido del fichero es XML. Esta decisión se toma para evitar que se puedan abrir accidentalmente archivos XML que no contengan rutas, aunque no se importara nada debido a su estructura interna. A continuación se muestra un ejemplo de este formato de archivo (los lugares no representan localizaciones reales).

```

<?xml version="1.0" encoding="UTF-8"?>
<lugares>
  <nombre_ruta>NombreRuta</nombre_ruta>
  <!-- Hay tantos elementos lugar como lugares tiene agregada la ruta -->
  <!-- Lugar no personalizado -->
  <lugar>
    <latitud>99.9999999</latitud>
    <longitud>99.9999999</longitud>
    <lugar_personalizado>no</lugar_personalizado>
  </lugar>
  <!-- Lugar personalizado -->
  <lugar>
    <latitud>01.9999999</latitud>
    <longitud>01.9999999</longitud>
    <lugar_personalizado>si</lugar_personalizado>
    <nombre>NombreLugar</nombre>
    <direccion>DireccionLugar</direccion>
    <localidad>LocalidadLugar</localidad>
    <provincia>ProvinciaLugar</provincia>
    <descripcion>DescripcionLugar</descripcion>
    <url>UrlLugar</url>
    <correo>CorreoLugar</correo>
    <telefono>TelefonoLugar</telefono>
    <descripcionLarga>DescripcionLargaLugar</descripcionLarga>
  </lugar>
</lugares>

```

## 7. Pruebas

### 7.1. Metodología utilizada

Como se explicó anteriormente, se fueron realizando pruebas a medida que se iba completando la funcionalidad de la aplicación, de manera incremental. Cada vez que se añadía una nueva característica, se probaba para comprobar su correcto funcionamiento, y los domingos de cada semana se realizaba una prueba general del sistema. Como se modificó el comportamiento de la aplicación a lo largo del desarrollo, las pruebas utilizadas se modifican, añadiéndose nuevas para casos que aparecían y eliminándose otras cuando no tenían validez. Las pruebas presentadas en esta sección son el resultado de la última ejecución durante la semana de pruebas de finales de marzo, en la que se comprobó que la aplicación funcionaba de forma satisfactoria, se corrigieron los últimos errores y se terminó el desarrollo de la aplicación.

Por su disponibilidad se decidió usar dispositivos reales para las pruebas, para comprobar el uso real de los servicios de localización (se usaron los dispositivos de varios familiares a los que se está agradecido por el préstamo). Además, se usó un dispositivo virtual para comprobar el funcionamiento de la aplicación en una versión de API no disponible en los dispositivos reales. Los dispositivos usados han sido:

- bq Aquaris E4 con Android 4.4 (API 19)
- bq Aquaris E4 con Android 5.0 (API 21)
- bq Aquaris 5 con Android 6.0 (API 23)
- Nexus 7 con Android 8.0 (Dispositivo virtual) (API 26)

A partir del nivel de API 23, Android modifica su política de permisos, permitiendo a los usuarios la concesión y la negación de permisos en tiempo de ejecución[26]. Por lo tanto, en aquellos casos de uso en los que difiere el comportamiento de la aplicación, se realizan tests paralelos para comprobar la funcionalidad en dispositivos con API menor que 23 y en dispositivos con API mayor o igual a 23. Los demás tests son independientes del nivel de API utilizado.

Las pruebas están agrupadas en función a los casos de uso a los que se aplica.

## 7.2. Resultado de las pruebas

### 7.2.1. Iniciar Aplicación

<b>Identificador de Prueba</b>	CU_01_01
<b>Descripción</b>	El usuario inicia la aplicación y no hay datos en la base de datos
<b>Resultado esperado</b>	Se muestra un mensaje informando de la situación y se solicita confirmación para descargar los datos
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_01_02
<b>Descripción</b>	El usuario cancela la descarga de los datos
<b>Resultado esperado</b>	No se activan las opciones de ActividadInicial
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_01_03
<b>Descripción</b>	El usuario confirma la descarga de los datos y está conectado a Internet
<b>Resultado esperado</b>	Se activa el ProgressBar, se descargan los datos, se informa al usuario y se desactiva el ProgressBar; incorporando a la base de datos los lugares de los ficheros
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_01_04
<b>Descripción</b>	El usuario confirma la descarga de los datos y no está conectado a Internet
<b>Resultado esperado</b>	Se muestra un mensaje informando de la situación
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_01_05
<b>Descripción</b>	El usuario inicia la aplicación y hay datos en la base de datos
<b>Resultado esperado</b>	Se activan las opciones de la actividad
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.2. Actualizar base de datos

<b>Identificador de Prueba</b>	CU_02_01
<b>Descripción</b>	El usuario selecciona la opción «Actualizar» y está conectado a Internet
<b>Resultado esperado</b>	Se activa el ProgressBar, se actualizan los datos, se informa al usuario y se desactiva el ProgressBar; incorporando a la base de datos los nuevos lugares, sin borrarse las rutas y sus itinerarios
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_02_02
<b>Descripción</b>	El usuario selecciona la opción «Actualizar» y no está conectado a Internet
<b>Resultado esperado</b>	Se informa al usuario y no se actualiza
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.3. Abrir Mapa

<b>Identificador de Prueba</b>	CU_03_01
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa» y no está conectado a Internet
<b>Resultado esperado</b>	El mapa aparece en blanco
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_03_02
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa», está conectado a Internet y tiene la localización activada
<b>Resultado esperado</b>	Se establecen los servicios de localización, se obtienen los lugares de la base de datos y se muestran en el mapa
<b>Nivel de API</b>	$\geq 19$ ; $< 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_03_03
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa», está conectado a Internet y no tiene la localización activada
<b>Resultado esperado</b>	No se establecen los servicios de localización, se obtienen los lugares de la base de datos y se muestran en el mapa
<b>Nivel de API</b>	$\geq 19$ ; $< 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_03_04
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa», está conectado a Internet, tiene la localización activada y ha dado los permisos
<b>Resultado esperado</b>	Se establecen los servicios de localización, se obtienen los lugares de la base de datos y se muestran en el mapa
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_03_05
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa», está conectado a Internet, tiene la localización activada y no ha dado los permisos
<b>Resultado esperado</b>	Se solicitan los permisos, se obtienen los lugares de la base de datos y se muestran en el mapa
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	<u>Fallo</u> : la aplicación se queda en bucle solicitando los permisos <u>Causa</u> : Al evaluar una condición se entra en bucle debido a la lógica interna de Android <u>Solución</u> : eliminar esa condición y cambiar el flujo al solicitar los permisos <u>Segunda ejecución</u> : Correcto

<b>Identificador de Prueba</b>	CU_03_04
<b>Descripción</b>	El usuario selecciona la opción «Abrir Mapa», está conectado a Internet, no tiene la localización activada y ha dado los permisos
<b>Resultado esperado</b>	No se establecen los servicios de localización, se obtienen los lugares de la base de datos y se muestran en el mapa
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	Correcto

#### 7.2.4. Agregar Lugar

<b>Identificador de Prueba</b>	CU_04_01
<b>Descripción</b>	Se mantiene pulsado en el mapa, se completa al menos el campo de nombre y se confirma
<b>Resultado esperado</b>	Se agrega el lugar a la base de datos y se actualiza el mapa
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	<u>Fallo</u> : la aplicación no actualiza el mapa mostrando el nuevo lugar <u>Causa</u> : No se refrescan los clusters <u>Solución</u> : refrescar los clusters al recargar el mapa <u>Segunda ejecución</u> : Correcto

<b>Identificador de Prueba</b>	CU_04_02
<b>Descripción</b>	Se mantiene pulsado en el mapa, se completa al menos el campo de nombre y se cancela
<b>Resultado esperado</b>	No se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_04_03
<b>Descripción</b>	Se mantiene pulsado en el mapa, no se completa el campo de nombre y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_04_04
<b>Descripción</b>	Se mantiene pulsado en el mapa, no se completa el campo de nombre y se cancela
<b>Resultado esperado</b>	No se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.5. Editar Lugar

<b>Identificador de Prueba</b>	CU_05_01
<b>Descripción</b>	Se selecciona la opción de edición de un lugar de tipo «Personalizado»
<b>Resultado esperado</b>	Se muestra en VistaLugar la información de ese Lugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_05_02
<b>Descripción</b>	Se selecciona la opción de edición de un lugar de tipo «Personalizado», no se modifica nada y se confirma
<b>Resultado esperado</b>	Se actualiza el lugar y se refresca el mapa
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_05_03
<b>Descripción</b>	Se selecciona la opción de edición de un lugar de tipo «Personalizado», se modifica al menos uno de los campos y se confirma
<b>Resultado esperado</b>	Se actualiza el Lugar y se refresca el mapa
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_05_04
<b>Descripción</b>	Se selecciona la opción de edición de un lugar de tipo «Personalizado», se borra el nombre y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_05_05
<b>Descripción</b>	Se cancela la edición
<b>Resultado esperado</b>	No se realiza ninguna acción y se vuelve a VistaLugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto



### 7.2.6. Eliminar Lugar

<b>Identificador de Prueba</b>	CU_06_01
<b>Descripción</b>	Se selecciona la opción de borrado de un lugar y no está agregado a ninguna ruta
<b>Resultado esperado</b>	Se elimina el lugar y se actualiza el mapa
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_06_02
<b>Descripción</b>	Se selecciona la opción de borrado de un lugar y está agregado a alguna ruta
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.7. Trazar Ruta a Lugar

<b>Identificador de Prueba</b>	CU_07_01
<b>Descripción</b>	Se selecciona la opción de trazar ruta a lugar, no hay ninguna ruta dibujada, se está conectado a Internet y se tiene activada la localización
<b>Resultado esperado</b>	Se dibuja la ruta entre la ubicación y el lugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_07_02
<b>Descripción</b>	Se selecciona la opción de trazar ruta a lugar, hay una ruta dibujada, se está conectado a Internet y se tiene activada la localización
<b>Resultado esperado</b>	Se borra la ruta anterior y se dibuja la ruta entre la ubicación y el lugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	<u>Fallo</u> : no se borra la ruta anterior si existe <u>Causa</u> : no se ha creado el método para borrar la ruta <u>Solución</u> : añadir el método y llamarlo <u>Segunda ejecución</u> : Correcto

<b>Identificador de Prueba</b>	CU_07_03
<b>Descripción</b>	Se selecciona la opción de trazar ruta a lugar, no hay ninguna ruta dibujada, no se está conectado a Internet y se tiene activada la localización
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_07_04
<b>Descripción</b>	Se selecciona la opción de trazar ruta a lugar, no hay ninguna ruta dibujada, se está conectado a Internet y no se tiene activada la localización
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_07_05
<b>Descripción</b>	Se selecciona la opción de trazar ruta a lugar, no hay ninguna ruta dibujada, se está conectado a Internet, se tiene activada la localización y no se encuentra la ruta
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.8. Mostrar Lugar en Mapa

<b>Identificador de Prueba</b>	CU_08_01
<b>Descripción</b>	Se selecciona la opción de mostrar lugar en mapa de un lugar
<b>Resultado esperado</b>	Se mueve la cámara hasta ese lugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.9. Llamar Número

<b>Identificador de Prueba</b>	CU_09_01
<b>Descripción</b>	Se pulsa sobre el número de teléfono de un lugar
<b>Resultado esperado</b>	Se llama a ese teléfono
<b>Nivel de API</b>	$\geq 19$ ; $< 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_09_02
<b>Descripción</b>	Se pulsa sobre el número de teléfono de un lugar y se han dado los permisos
<b>Resultado esperado</b>	Se llama a ese teléfono
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_09_03
<b>Descripción</b>	Se pulsa sobre el número de teléfono de un lugar y no se han dado los permisos
<b>Resultado esperado</b>	Se solicitan los permisos
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	Correcto

### 7.2.10. Abrir Correo

<b>Identificador de Prueba</b>	CU_10_01
<b>Descripción</b>	Se pulsa sobre el correo de un lugar
<b>Resultado esperado</b>	Se abre el gestor de correo elegido por el usuario, con esa dirección de correo como destinatario
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.11. Abrir web

<b>Identificador de Prueba</b>	CU_11_01
<b>Descripción</b>	Se pulsa sobre la web de un lugar
<b>Resultado esperado</b>	Se abre el navegador web elegido por el usuario, abriendo la dirección web del lugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.12. Mostrar Información de Lugar

<b>Identificador de Prueba</b>	CU_12_01
<b>Descripción</b>	Se pulsa sobre la ventana de información del marcador de un lugar
<b>Resultado esperado</b>	Se muestra la información del lugar en VistaLugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_12_02
<b>Descripción</b>	Se pulsa sobre un elemento de ListaLugares
<b>Resultado esperado</b>	Se muestra la información del lugar en VistaLugar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_12_03
<b>Descripción</b>	El Lugar no tiene alguno de los campos
<b>Resultado esperado</b>	Ese campo aparece como «No disponible»
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	<u>Fallo</u> : al cambiar entre traducciones algunos valores no se muestran correctamente <u>Causa</u> : no se han traducido todas las cadenas de texto <u>Solución</u> : buscar y traducir todos las cadenas de texto <u>Segunda ejecución</u> : Correcto

### 7.2.13. Mostrar Rutas

<b>Identificador de Prueba</b>	CU_13_01
<b>Descripción</b>	Se selecciona la opción de mostrar rutas, existen rutas en la aplicación
<b>Resultado esperado</b>	Se muestran las rutas en ListaRutas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_13_02
<b>Descripción</b>	Se selecciona la opción de mostrar rutas y no existen rutas en la aplicación
<b>Resultado esperado</b>	Se informa al usuario y se muestra ListaRutas vacío
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.14. Ver Ruta

<b>Identificador de Prueba</b>	CU_14_01
<b>Descripción</b>	Se pulsa sobre una ruta y la ruta tiene lugares agregados
<b>Resultado esperado</b>	Se muestran los lugares del itinerario en ListaLugares
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_14_02
<b>Descripción</b>	Se pulsa sobre una ruta y la ruta no tiene lugares agregados
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.15. Agregar Ruta

<b>Identificador de Prueba</b>	CU_15_01
<b>Descripción</b>	Se selecciona la opción de agregar ruta, se introduce un nombre nuevo y se confirma
<b>Resultado esperado</b>	Se crea la ruta y se actualizan las rutas a mostrar
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_15_02
<b>Descripción</b>	Se selecciona la opción de agregar ruta, se introduce un nombre ya existente y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	<u>Fallo</u> : la aplicación se detiene <u>Causa</u> : no se comprueba si ya existe una ruta con ese nombre, provocando una excepción que detiene la aplicación <u>Solución</u> : comprobar si existe una ruta con ese nombre <u>Segunda ejecución</u> : Correcto

<b>Identificador de Prueba</b>	CU_15_03
<b>Descripción</b>	Se selecciona la opción de agregar ruta, no se introduce un nombre y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_15_04
<b>Descripción</b>	Se selecciona la opción de agregar ruta y se cancela
<b>Resultado esperado</b>	No se realiza ninguna acción y se vuelve a ListaRutas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.16. Editar Nombre de Ruta

<b>Identificador de Prueba</b>	CU_16_01
<b>Descripción</b>	Se selecciona la opción de edición de nombre de una ruta, se introduce un nuevo nombre y se confirma
<b>Resultado esperado</b>	Se actualiza la ruta y se actualizan las rutas mostradas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_16_02
<b>Descripción</b>	Se selecciona la opción de edición de nombre de una ruta, se introduce un nombre ya existente y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_16_03
<b>Descripción</b>	Se selecciona la opción de edición de nombre de una ruta, se borra el nombre y se confirma
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_16_04
<b>Descripción</b>	Se selecciona la opción de edición de nombre de una ruta y se cancela
<b>Resultado esperado</b>	No se realiza ninguna acción y se vuelve a ListaRutas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

#### 7.2.17. Eliminar Ruta

<b>Identificador de Prueba</b>	CU_17_01
<b>Descripción</b>	Se selecciona la opción de eliminar ruta y se confirma
<b>Resultado esperado</b>	Se elimina la ruta y se actualizan las rutas mostradas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_17_02
<b>Descripción</b>	Se selecciona la opción de eliminar ruta y se cancela
<b>Resultado esperado</b>	No se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

#### 7.2.18. Mostrar Ruta en Mapa

<b>Identificador de Prueba</b>	CU_18_01
<b>Descripción</b>	Se selecciona la opción de mostrar ruta en mapa, se está conectado a Internet, no hay dibujada ninguna ruta y se tiene la localización activada
<b>Resultado esperado</b>	Se dibuja la ruta entre la localización del usuario y el primer lugar y entre los lugares de la ruta
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_18_02
<b>Descripción</b>	Se selecciona la opción de mostrar ruta en mapa, se está conectado a Internet, hay dibujada una ruta y se tiene la localización activada
<b>Resultado esperado</b>	Se borra la ruta anterior y se dibuja la ruta entre la localización del usuario y el primer lugar y entre los lugares de la ruta
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_18_03
<b>Descripción</b>	Se selecciona la opción de mostrar ruta en mapa, no se está conectado a Internet, no hay dibujada ninguna ruta y se tiene la localización activada
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_18_04
<b>Descripción</b>	Se selecciona la opción de mostrar ruta en mapa, se está conectado a Internet, no hay dibujada ninguna ruta y no se tiene la localización activada
<b>Resultado esperado</b>	Se dibuja la ruta entre los lugares de la ruta
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_18_05
<b>Descripción</b>	Se selecciona la opción de mostrar ruta en mapa, se está conectado a Internet, no hay dibujada ninguna ruta, se tiene la localización activada y no se encuentra alguna ruta
<b>Resultado esperado</b>	Se dibujan las rutas encontradas
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto



## 7.2.19. Editar Ruta

<b>Identificador de Prueba</b>	CU_19_01
<b>Descripción</b>	Se selecciona una ruta para editar
<b>Resultado esperado</b>	Se activa el modo de edición y se cambia el color
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	<p><u>Fallo</u>: la aplicación se detiene</p> <p><u>Causa</u>: no se realiza el cambio de color correctamente, provocando una excepción</p> <p><u>Solución</u>: modificar la forma en la que se cambia el color</p> <p><u>Segunda ejecución</u>: Correcto</p>

<b>Identificador de Prueba</b>	CU_19_02
<b>Descripción</b>	Se selecciona una ruta para editar, se agregan varios lugares y se confirman los cambios
<b>Resultado esperado</b>	Se agregan los lugares a la ruta y se informa al usuario de la finalización de la edición
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_19_03
<b>Descripción</b>	Se selecciona una ruta para editar, no se agregan lugares y se confirman los cambios
<b>Resultado esperado</b>	Se informa al usuario de la finalización de la edición
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_19_04
<b>Descripción</b>	Se selecciona una ruta para editar, se agregan lugares y se cancelan los cambios
<b>Resultado esperado</b>	Se informa al usuario de la finalización de la edición
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_19_05
<b>Descripción</b>	Se pulsa «Cancelar» o «Listo!»
<b>Resultado esperado</b>	Se desactiva el modo de edición y se cambia el color
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

## 7.2.20. Compartir Ruta

<b>Identificador de Prueba</b>	CU_20_01
<b>Descripción</b>	Se selecciona una ruta para compartir, se tienen permisos de escritura y está montado el almacenamiento externo
<b>Resultado esperado</b>	Se escribe el fichero con la ruta y se abre el selector de medios
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_20_02
<b>Descripción</b>	Se selecciona una ruta para compartir, no se tienen permisos de escritura y está montado el almacenamiento externo
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$ ; $< 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_20_03
<b>Descripción</b>	Se selecciona una ruta para compartir, se tienen permisos de escritura y no está montado el almacenamiento externo
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_20_04
<b>Descripción</b>	Se selecciona una ruta para compartir, no se tienen permisos de escritura y está montado el almacenamiento externo
<b>Resultado esperado</b>	Se solicitan los permisos
<b>Nivel de API</b>	$\geq 23$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_20_05
<b>Descripción</b>	Ocurre un fallo durante la escritura del fichero
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.21. Importar Ruta

<b>Identificador de Prueba</b>	CU_21_01
<b>Descripción</b>	Se selecciona la opción de importar ruta, se tienen permisos de lectura del almacenamiento externo, se escoge un fichero que sigue el formato de nombre y es de tipo «.intour» y el nombre de la ruta no está en la base de datos
<b>Resultado esperado</b>	Se agrega la Ruta a la base de datos
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	<u>Fallo</u> : no se importa la ruta <u>Causa</u> : dependiendo del gestor de ficheros elegido la ruta se puede devolver en diferentes formatos <u>Solución</u> : modificar la forma en la que se obtiene la ruta del fichero <u>Segunda ejecución</u> : Correcto

<b>Identificador de Prueba</b>	CU_21_02
<b>Descripción</b>	Se selecciona la opción de importar ruta, no se tienen permisos de lectura del almacenamiento externo
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	≥ 19; < 23
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_21_03
<b>Descripción</b>	Se selecciona la opción de importar ruta, no se tienen permisos de lectura del almacenamiento externo
<b>Resultado esperado</b>	Se solicitan permisos
<b>Nivel de API</b>	≥ 23
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_21_04
<b>Descripción</b>	El archivo seleccionado no sigue el formato de nombre
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_21_05
<b>Descripción</b>	El archivo seleccionado no es de tipo «.intour»
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	≥ 19
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_21_06
<b>Descripción</b>	El nombre de la ruta ya existe en la base de datos
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_21_07
<b>Descripción</b>	La ruta contiene lugares personalizados
<b>Resultado esperado</b>	Se agrega la Ruta a la base de datos y se crean los lugares personalizados
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.22. Mostrar Lugares Cercanos

<b>Identificador de Prueba</b>	CU_22_01
<b>Descripción</b>	Se selecciona la opción de mostrar lugares cercanos, se tiene activada la localización y hay lugares visibles dentro del radio
<b>Resultado esperado</b>	Se muestran los lugares
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_22_02
<b>Descripción</b>	Se selecciona la opción de mostrar lugares cercanos, no se tiene activada la localización y hay lugares visibles dentro del radio
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_22_03
<b>Descripción</b>	Se selecciona la opción de mostrar lugares cercanos, se tiene activada la localización y no hay lugares visibles dentro del radio
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

### 7.2.23. Realizar búsqueda de lugares

<b>Identificador de Prueba</b>	CU_23_01
<b>Descripción</b>	Se introduce texto para buscar coincidencias, se encuentran resultados y son visibles
<b>Resultado esperado</b>	Se muestran los resultados
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_23_02
<b>Descripción</b>	Se introduce texto para buscar coincidencias, no se encuentran resultados
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

<b>Identificador de Prueba</b>	CU_23_03
<b>Descripción</b>	Se introduce texto para buscar coincidencias, se encuentran resultados y no son visibles
<b>Resultado esperado</b>	Se informa al usuario y no se realiza ninguna acción
<b>Nivel de API</b>	$\geq 19$
<b>Resultado</b>	Correcto

## 8. Conclusiones y Trabajo Futuro

A lo largo del presente Trabajo de Fin de Grado, se ha podido comprobar la facilidad para el desarrollo en dispositivos móviles basados en Android, así como su nada desdeñable potencia que no se ve lastrada por esta sencillez. A pesar de que al principio, puede resultar algo desalentador al presentarse una arquitectura un tanto peculiar e incorporar bastante funcionalidad propia basada en Java, en apenas unas sesiones de aprendizaje se dominan los conceptos clave; a partir de ese momento, no se hace nada más que incrementar los conocimientos y el dominio de la plataforma. Durante toda la fase de desarrollo se ha podido comprobar de primera mano, la razón por la cual actualmente Android es la plataforma líder en dispositivos móviles.

Otro aspecto experimentado durante el desarrollo ha sido el de crear una aplicación que tuviera que funcionar en una amplia gama de dispositivos, con diferentes características y en entornos con diferencias notables. Por ejemplo, al utilizar los servicios de localización, se tiene que tener en cuenta si el dispositivo tiene una API mayor al nivel 23, ya que si es así, los permisos se pueden rescindir en tiempo de ejecución. Esto involucra tener que comprobar si se cuenta con los permisos antes de realizar cualquier operación sobre la ejecución. Otro aspecto de este paradigma es tener que diseñar la interfaz completamente independiente del dispositivo, y tener en cuenta los tamaños de pantalla, sus densidades y su resolución; un aspecto claramente opuesto a la filosofía del principal competidor de Android, Apple.

Igualmente se ha podido comprobar, los cambios que puede llegar a experimentar Android en apenas un año, por ejemplo, con la incorporación de Android Room, que elimina el uso directo, un tanto complicado a veces, de la API de SQLite.

A mayores se ha trabajado integrando diferentes APIs independientes, ya sea para obtener la localización, calcular rutas o mostrar el mapa. Además, se utilizaron ficheros externos de organismos oficiales, en este caso la Junta de Castilla y León. Estos dos factores hicieron otra vez, el tener que tener en cuenta todas las posibles situaciones de fallo para garantizar el correcto funcionamiento de la aplicación.

Nunca antes hasta la realización del TFG se habían realizado pruebas tan exhaustivas sobre una aplicación, ni se había contado con usuarios reales que proporcionaran realimentación, provocando cambios debido a sus sugerencias. Sin lugar a dudas, ha sido una experiencia positiva, de la que se han extraído valiosas lecciones.

En último lugar, éste ha sido el desarrollo más largo al que el autor del TFG se ha enfrentado a lo largo de la carrera, y en el que más tiempo ha invertido, tanto diseñando y desarrollando, como realizando toda la documentación que debe acompañar a un proyecto de estas características. Los conocimientos adquiridos a lo largo de este periodo han demostrado ser útiles y completamente

válidos, ya fuera programando en Java como realizando diagramas de secuencia en astah. Por lo tanto, se puede concluir que se llega al final con la convicción de haber cumplido los objetivos propuestos de forma provechosa y satisfecho con todos los nuevos conocimientos y habilidades adquiridos.

Como líneas de trabajo futuro se incluyen las siguientes. La mayor parte de ellas surgen a través de la experiencia de usuarios reales con la aplicación (a los que se agradece el tiempo dedicado a ello), mientras que otras son fruto de un intento de mejora de la usabilidad.

- Establecer iconos más explicativos a los marcadores de los lugares, sustituyendo el código actual de color.
- Permitir eliminar lugares de una ruta mientras se esté editando. Actualmente sólo se permite agregar, pero no eliminar de una ruta.
- Al agregar un lugar personalizado, y si no ha sido posible encontrar la dirección por medio del GeoCoder, permitir al usuario agregarla manualmente.
- Al trazar una ruta, si entre dos lugares no se encuentra la ruta, intentar buscar un camino alternativo (por ejemplo, entre lugar1 y lugar2; lugar3 y lugar4 se encuentra, pero no entre lugar2 y lugar3. Se intentarían buscar alternativas, como lugar3 y lugar1 o lugar4 y lugar1, etc).
- Actualmente se busca la ruta a partir del orden de aparición de los lugares en ella. Una mejora sería buscar el recorrido óptimo entre los lugares (más corto, más rápido, etc).
- Al modificarse las preferencias de ruta y si hay trazada una ruta, actualizar la ruta a los nuevos valores.
- Permitir al usuario definir preferencias del proveedor de GPS.
- Asociar los archivos con formato «.intour» a la aplicación, y permitir que al abrir un archivo se importe automáticamente.

## 9. Referencias

- [1] Abel y col. *Tratamiento de XML en Android (I): SAX*. Jun. de 2013. URL: <http://www.sgoliver.net/blog/tratamiento-de-xml-en-android-i-sax/> (visitado 09-02-2018).
- [2] Sohail Aziz. *Little things about android and security*. Oct. de 2014. URL: <http://sohailaziz05.blogspot.com.es/2014/10/asynctask-and-context-leaking.html> (visitado 22-02-2018).
- [3] Europa Press. *El empleo turístico en España*. Nov. de 2016. URL: <https://www.20minutos.es/noticia/2886394/0/empleo-turistico-espana-25-millones-trabajadores-epa/> (visitado 12-03-2018).
- [4] Gironés Jesús Tomás. *El Gran libro de Android*. Marcombo, 2016.
- [5] Florina Muntelescu. *7 Pro-tips for Room – Google Developers – Medium*. Nov. de 2017. URL: <https://medium.com/google-developers/7-pro-tips-for-room-fbadea4bfbd1> (visitado 19-02-2018).
- [6] *Accessing data using Room DAOs*. Feb. de 2018. URL: <https://developer.android.com/training/data-storage/room/accessing-data.html> (visitado 15-02-2018).
- [7] *Adding an Action to a Message*. Mar. de 2018. URL: <https://developer.android.com/training/snackbar/action.html> (visitado 07-04-2018).
- [8] *AsyncTask*. Abr. de 2018. URL: <https://developer.android.com/reference/android/os/AsyncTask.html> (visitado 08-04-2018).
- [9] Castilla. *El crecimiento turístico en Castilla y León alcanza en 2017 cifras históricas de empleo en el sector y supera por primera vez los 8 millones de turistas*. 2018. URL: [https://comunicacion.jcyl.es/web/jcyl/Comunicacion/es/Plantilla100Detalle/1284663638052/\\_/1284777853017/Comunicacion](https://comunicacion.jcyl.es/web/jcyl/Comunicacion/es/Plantilla100Detalle/1284663638052/_/1284777853017/Comunicacion) (visitado 12-03-2018).
- [10] *Create, read, update and delete*. Abr. de 2018. URL: [https://en.wikipedia.org/wiki/Create,\\_read,\\_update\\_and\\_delete](https://en.wikipedia.org/wiki/Create,_read,_update_and_delete) (visitado 01-04-2018).
- [11] *Data access object*. Abr. de 2018. URL: [https://en.wikipedia.org/wiki/Data\\_access\\_object](https://en.wikipedia.org/wiki/Data_access_object) (visitado 01-04-2018).
- [12] *Defining data using Room entities*. Feb. de 2018. URL: <https://developer.android.com/training/data-storage/room/defining-data.html> (visitado 15-02-2018).
- [13] *Entity*. Feb. de 2018. URL: <https://developer.android.com/reference/android/arch/persistence/room/Entity.html> (visitado 15-02-2018).
- [14] *PreferenceFragment*. Abr. de 2018. URL: <https://developer.android.com/reference/android/preference/PreferenceFragment.html> (visitado 08-04-2018).
- [15] *Relational model*. Abr. de 2018. URL: [https://en.wikipedia.org/wiki/Relational\\_model](https://en.wikipedia.org/wiki/Relational_model) (visitado 01-04-2018).



- [16] *Save data in a local database using Room*. Feb. de 2018. URL: <https://developer.android.com/training/data-storage/room/index.html> (visitado 15-02-2018).
- [17] *Toasts*. Mar. de 2018. URL: <https://developer.android.com/guide/topics/ui/notifiers/toasts.html> (visitado 07-04-2018).
- [18] CommonsWare. *Room and Custom Types*. URL: <https://commonsware.com/AndroidArch/previews/room-and-custom-types> (visitado 16-02-2018).
- [19] *Distribution dashboard*. URL: <https://developer.android.com/about/dashboards/> (visitado 01-02-2018).
- [20] Emmax. *MVC vs. MVP vs. MVVM on Android*. URL: <https://academy.realm.io/posts/eric-maxwell-mvc-mvp-and-mvvm-on-android/> (visitado 13-04-2018).
- [21] *Geocoder*. URL: <https://developer.android.com/reference/android/location/Geocoder> (visitado 21-04-2018).
- [22] *Google Maps API Documentation*. URL: <https://developers.google.com/maps/documentation/android-api/start?hl=es-419> (visitado 17-02-2018).
- [23] *Google Maps Directions API Documentation*. URL: <https://developers.google.com/maps/documentation/directions/start?hl=es-419> (visitado 03-02-2018).
- [24] *LocationManager*. URL: <https://developer.android.com/reference/android/location/LocationManager> (visitado 08-04-2018).
- [25] *Mobile Operating System Market Share Worldwide*. URL: <http://gs.statcounter.com/os-market-share/mobile/worldwide> (visitado 01-02-2018).
- [26] *Permissions Overview*. URL: <https://developer.android.com/guide/topics/permissions/overview> (visitado 15-04-2018).

## A. Distribución de contenidos del soporte digital

Dentro de la memoria USB se encuentra una carpeta llamada «Trabajo de Fin de Grado», esta es la carpeta raíz del trabajo. Dentro de ella se encuentran las siguientes carpetas y archivos:

- InfoTouristTFG: carpeta con el proyecto de Android Studio de la aplicación.
- Imágenes: carpeta con todas las imágenes de los diagramas utilizados en la memoria
- InfoTourist.ast: archivo Astah con todos los diagramas utilizados en la aplicación
- Memoria.pdf: la memoria del trabajo.
- InfoTourist-1.0-release.apk: Ejecutable de la aplicación

## B. Manual de Instalación

Requisitos:

- Teléfono móvil con Android 4.4 (API 19) o superior. En caso de no disponer de un dispositivo físico se puede usar el emulador incorporado en Android Studio, aunque se deben tener en cuenta sus requisitos en cuanto a hardware.
- Tener en la memoria del dispositivo (interna o externa) el archivo InfoTourist-1.0-release.apk.
- Tener habilitada la opción de Android «Orígenes desconocidos».

Pasos a seguir

1. Localizar el archivo InfoTourist-1.0-release.apk y pulsar sobre él.
2. Seleccionar el gestor de instalación de paquetes deseado.
3. Esperar a que termine el proceso de instalación.

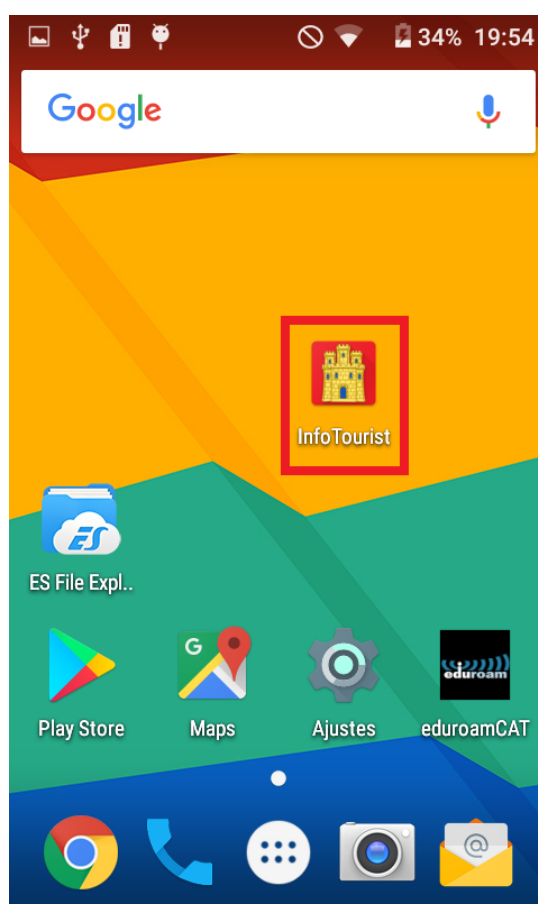
## C. Manual de Usuario

### C.1. Descripción

Este manual detalla la manera en la que se usa la aplicación InfoTourist, a través de imágenes en las que se especifica las diferentes opciones a las que pueden acceder los usuarios. Debe tenerse en cuenta que estas imágenes han sido tomadas desde un bq Aquaris e4 con Android 5.0, en posición vertical, por lo tanto la localización exacta de las opciones puede variar, aunque en absoluto su comportamiento. Además, el idioma usado ha sido español, en los otros idiomas el texto se muestra diferente. En relación a los permisos, en dispositivos con Android 6.0 o mayor, se mostrará un mensaje y se solicitará confirmación si no se tienen los permisos.

### C.2. Iniciar la aplicación

Para iniciar la aplicación, ésta debe estar instalada en el dispositivo, en la sección Manual de instalación se explica este procedimiento. Una vez instalada, para iniciarla simplemente se pulsa sobre su icono. Cuando esté abierta se puede acceder a toda su funcionalidad.



### C.3. Pantalla inicial

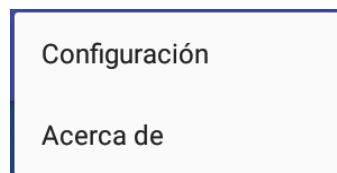


**Figura 46:** Pantalla Inicial

Esta es la pantalla inicial de la aplicación, si se pulsa sobre «Iniciar» se inicia Mapa donde se muestran los lugares.

La barra de aplicación de la pantalla dispone de varias opciones:

- Actualizar: actualiza los datos de la aplicación.
- Acceso a menú: se despliega el menú de la aplicación.



**Figura 47:** Menú de la pantalla Inicial

Desde el menú se puede acceder a varias opciones:

- Configuración: se accede a las Preferencias de la aplicación.
- Acerca de: muestra información sobre la aplicación en Acerca de.

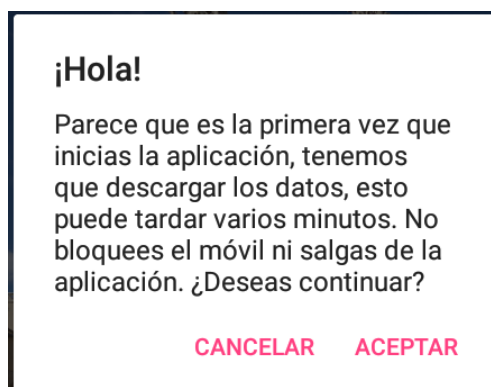
*Nota: este comportamiento es común a todas las pantallas, dependiendo de la longitud de la barra de aplicación, aparecerán más o menos opciones. De ahora en adelante se explicará la funcionalidad en conjunto.*

*Nota: el acceso a Configuración y a AcercaDe es el mismo en todas las pantallas donde se permite su opción.*

Cuando se descargan los datos por primera vez o se actualizan, la pantalla se muestra de la siguiente manera.



Si es la primera vez que se abre la aplicación se muestra este mensaje



Si se pulsa sobre actualizar se muestra este mensaje

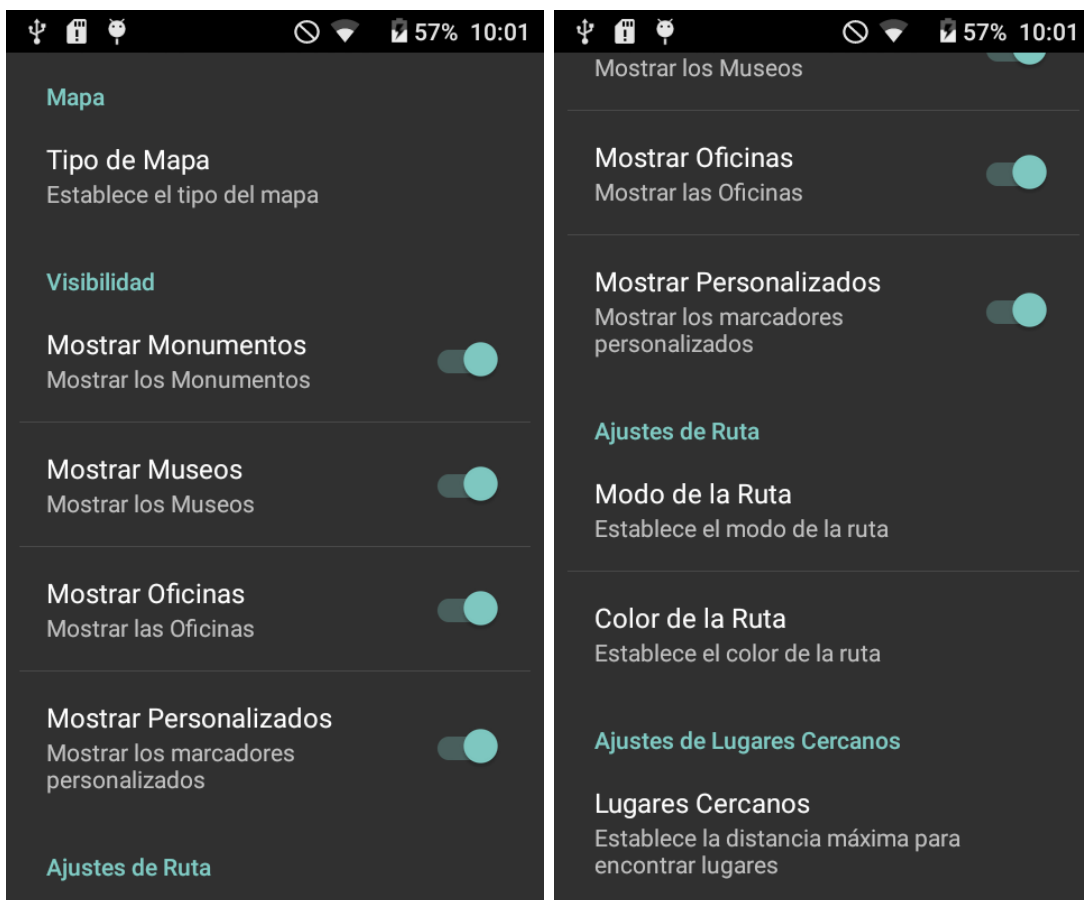
## Actualizar

Ahora actualizaremos los datos, esto puede tardar varios minutos. No bloquee el móvil ni salgas de la aplicación. ¿Deseas continuar?

CANCELAR ACEPTAR

## C.4. Preferencias

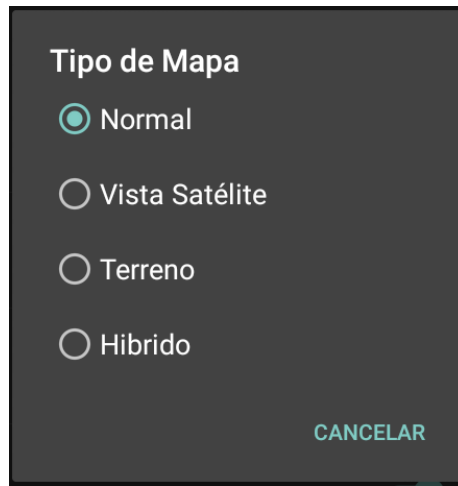
La pantalla de preferencias se ve de la siguiente forma, hay que desplazarse hacia abajo para ver todas las opciones:



En esta pantalla se encuentran las siguientes opciones:

### C.4.1. Tipo de mapa

Establece el tipo de mapa. Se pueden escoger los valores de vista normal, vista satélite, terreno (con las elevaciones en 3d) e híbrido (una mezcla de los dos anteriores).

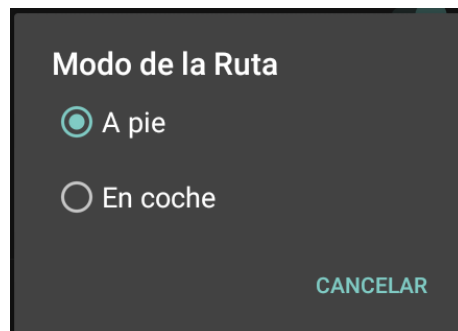


#### C.4.2. Visibilidad

Activa y/o desactiva la visibilidad de los diferentes tipos de lugares, cuando se desactiva la visibilidad, esos lugares no se mostrarán en el mapa, ni al buscar coincidencias ni al mostrar los lugares cercanos; pero sí se ven en las rutas. Para activar/desactivar se debe pulsar sobre el icono de la derecha.

#### C.4.3. Modo de la Ruta

Establece el modo en que se calcula la ruta. Se pueden escoger los valores a pie o en coche.



#### C.4.4. Color de la ruta

Establece el color en el que se dibuja la ruta. Se pueden escoger los valores de rojo, verde, azul, amarillo, negro y blanco.

**Color de la Ruta**

Rojo

Verde

Azul

Amarillo

Blanco

Negro

CANCELAR

#### C.4.5. Lugares Cercanos

Establece la distancia máxima a la que buscar lugares cercanos. Se pueden escoger los valores de 500m, 1km, 2km, 3km y 5km.

**Lugares Cercanos**

500 m

1 Km

2 Km

3 Km

5 Km

CANCELAR



## C.5. Mapa

Esta pantalla muestra los lugares que se encuentran en la base de datos.



**Figura 48:** Pantalla Mapa

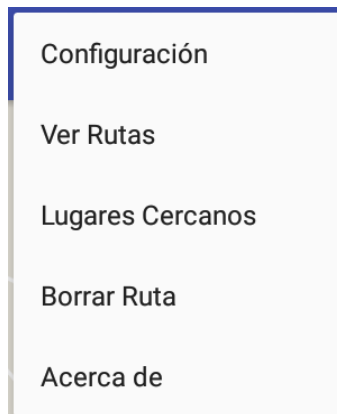
En esta pantalla se muestran las siguientes opciones:

1. Botones para acercar/alejar la cámara
2. Icono de marcador
3. Botón para ir a la ubicación del usuario (no se encuentra disponible si no se dan los permisos adecuados)
4. Buscador de coincidencias de texto, muestra los resultados en Lista Lugares

Si no se tiene la ubicación activada, se muestra este lugar por defecto, si está activada la cámara se desplaza hasta la ubicación del usuario (el círculo azul del centro de la pantalla)



Las opciones de menú son las siguientes:



- Configuración: abre la pantalla de Preferencias.
- Ver Rutas: muestra las rutas que contiene la aplicación en Lista Rutas
- Lugares Cercanos: muestra los lugares cercanos en Lista Lugares
- Borrar Ruta: si hay una ruta dibujada en el mapa, se borra
- Acerca de: muestra la pantalla Acerca de

Los lugares están identificados por un código de color:



Si se pulsa sobre un lugar, se activan las siguientes opciones:



1. Ventana de información del lugar
2. Nombre del lugar
3. Tipo del lugar
4. Botón para trazar la ruta, al lugar desde la ubicación (si no se encuentra disponible aparecerá un mensaje)

Si se pulsa sobre la ventana de información, se navega hasta la pantalla Información de Lugar.

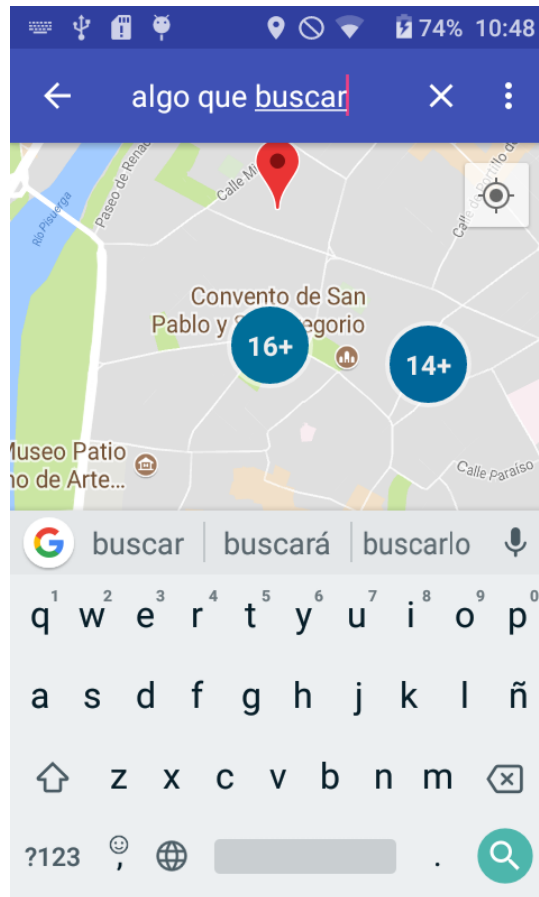
Si se mantiene pulsado sobre una posición del mapa, se abre la pantalla de Edición de Lugar. Si se confirman los cambios, el mapa se actualiza mostrando el nuevo lugar añadido.

Si se aleja la cámara lo suficiente, los lugares se agrupan en clusters, impidiendo de esta forma que la pantalla se sature de marcadores. Cada cluster tiene un número, que indica el número de lugares que agrupa. Para volver a ver los lugares se puede volver a acercar la cámara o pulsar sobre el cluster (se requiere pulsar varias veces, dependiendo de los lugares que tenga el cluster)



Al pulsarse sobre Lugares Cercanos se muestran los lugares que estén a igual o menor distancia que la definida en las preferencias, abriéndose la pantalla Lista Lugares, si se encuentran, en caso contrario se informa.

Al pulsarse sobre el buscador, se abre un cuadro de texto donde se introduce la(s) palabra(s) a buscar. Una vez realizada la búsqueda se muestran en Lista Lugares los resultados, si no se encuentran se informa.

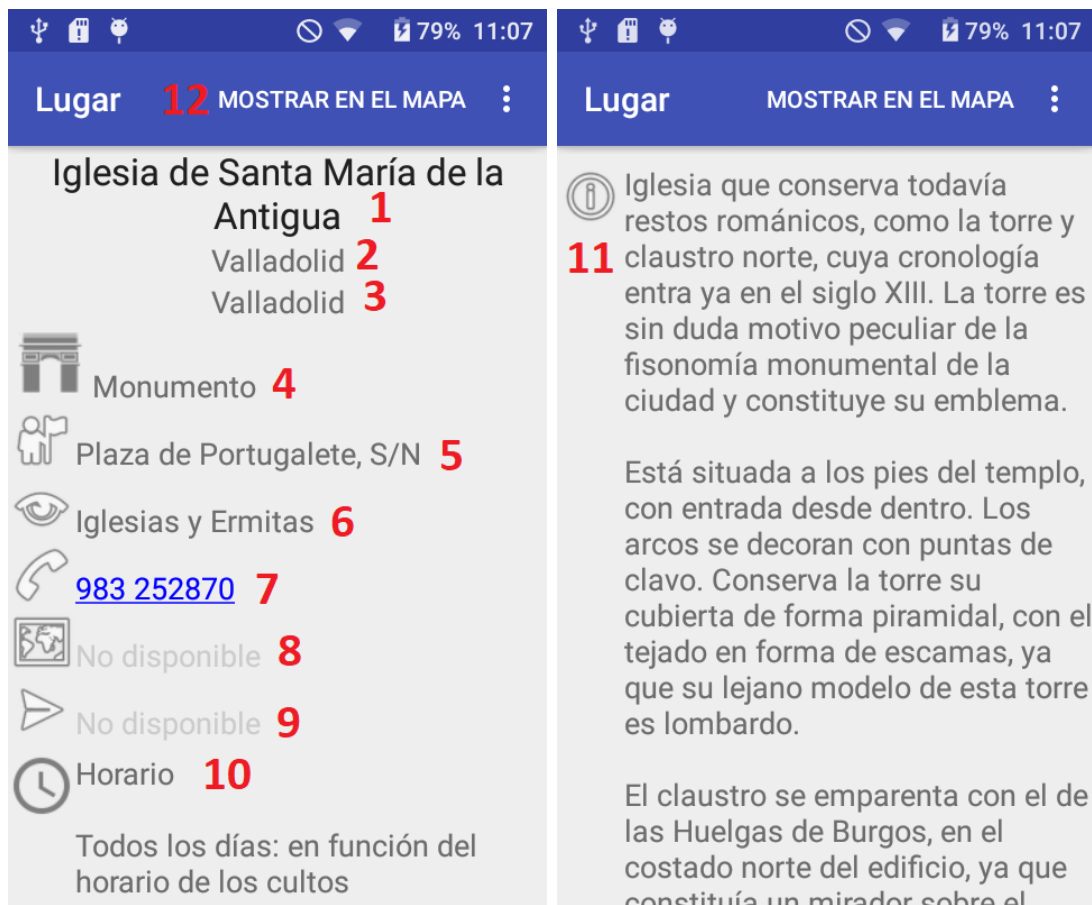


Al pulsarse sobre «Llévame hasta aquí» se traza la ruta hasta el lugar.



## C.6. Información de Lugar

Esta pantalla muestra la información de un lugar.



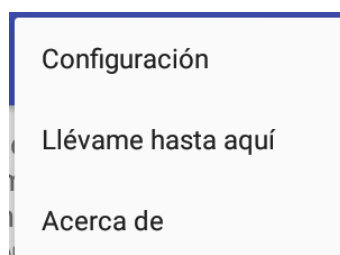
En ella podemos encontrar las siguientes opciones:

1. Nombre del lugar
2. Localidad del lugar
3. Provincia del lugar
4. Tipo del lugar. Cada icono representa un tipo:
  - Arco del triunfo: Monumento
  - Templo griego: Museo
  - Círculo con la letra «i»: Oficina de Información Turística
  - Marcador: Personalizado
5. Dirección del lugar
6. Subtipo del lugar en el caso de ser monumento

7. Teléfono del lugar, si está disponible, se puede llamar pulsando sobre el número
8. Web del lugar, si está disponible, se puede abrir en el navegador pulsando sobre la dirección
9. Correo del lugar, si está disponible, se puede escribir un correo a esa dirección pulsando sobre la dirección.
10. Horarios y Tarifas del lugar
11. Descripción del lugar
12. Mostrar en mapa: muestra en Mapa el lugar, centrado en la pantalla y con la ventana de información abierta

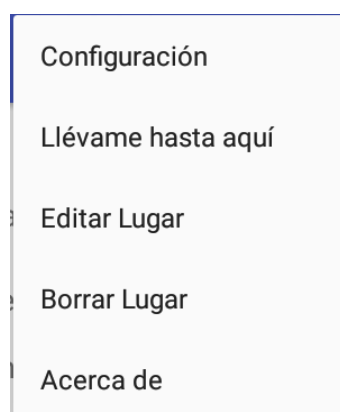
Si alguno de los campos anteriores no se encuentra se completa con «No disponible». Si se encuentran, Teléfono, Web y Correo aparecen en azul y subrayados.

En el menú se encuentran las siguientes opciones:



- Configuración: abre la pantalla de Preferencias
- Llévame hasta aquí: dibuja en Mapa la ruta hasta ese lugar, si no se encuentra se informa.
- Acerca de: abre la pantalla de Acerca de

Si el lugar es de tipo «Personalizado», se añaden dos nuevas opciones al menú:

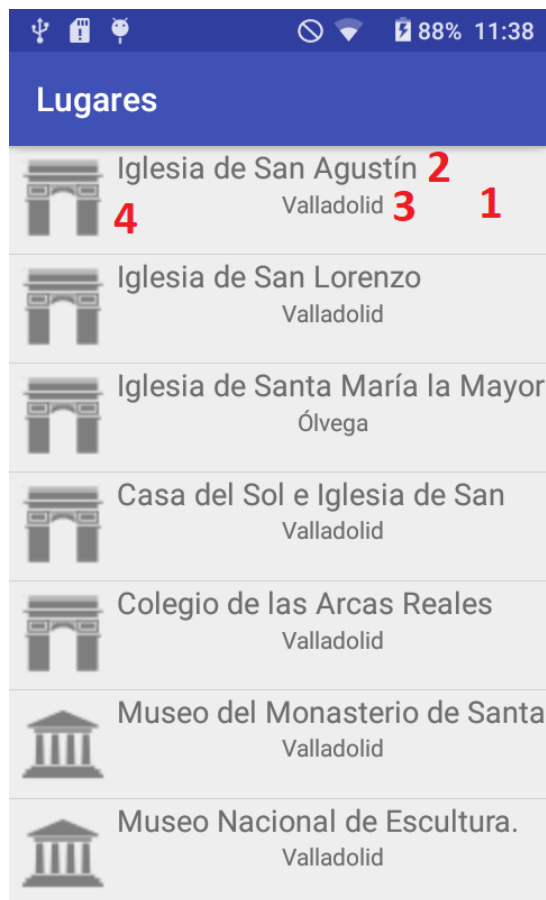




- Editar Lugar: abre la pantalla de Edición de Lugar y permite editar el lugar. Una vez hecho se actualiza su información.
- Borrar Lugar: elimina el lugar, si está agregado a una ruta se informa y no se borra.

## C.7. Lista Lugares

Esta pantalla se muestra cada vez que es necesario mostrar una lista de lugares (al devolver los resultados de la búsqueda de coincidencias, al mostrar los lugares cercanos o al ver los lugares de una ruta)



**Figura 49:** Lista Lugares

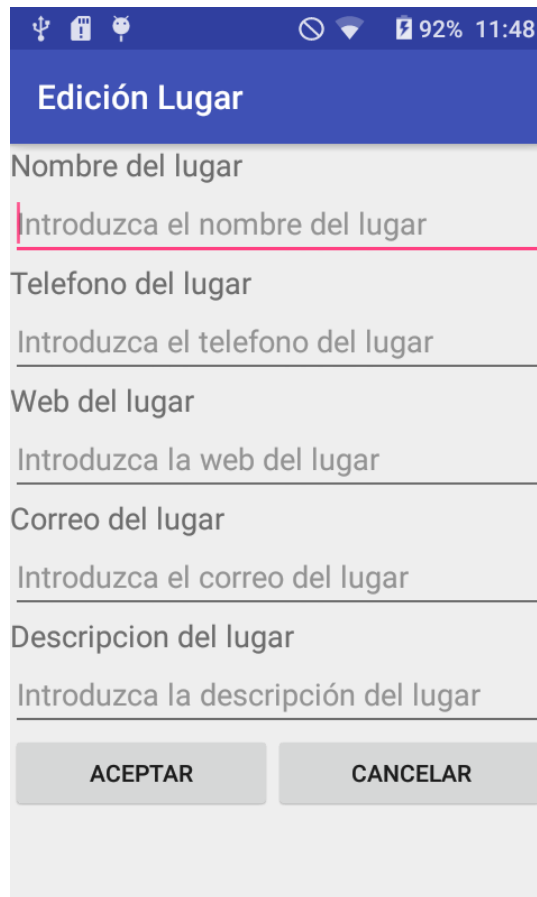
Se encuentran los siguientes elementos:

1. Elemento de la lista. Al pulsar sobre uno de ellos se abre la pantalla Información de lugar.
2. Nombre del lugar
3. Localidad del lugar
4. Icono que especifica el tipo del lugar

Se puede desplazar la pantalla hacia arriba y hacia abajo para ver todos los resultados.

## C.8. Edición de Lugar

Esta pantalla se muestra cuando se agrega/edita un lugar Personalizado. Dependiendo de esto se mostrarán los campos vacíos o no. El campo de nombre aparece subrayado al ser el más importante.



Edición Lugar

Nombre del lugar  
Introduzca el nombre del lugar

Telefono del lugar  
Introduzca el telefono del lugar

Web del lugar  
Introduzca la web del lugar

Correo del lugar  
Introduzca el correo del lugar

Descripcion del lugar  
Introduzca la descripción del lugar

ACEPTAR CANCELAR

**Figura 50:** Agregando Lugar

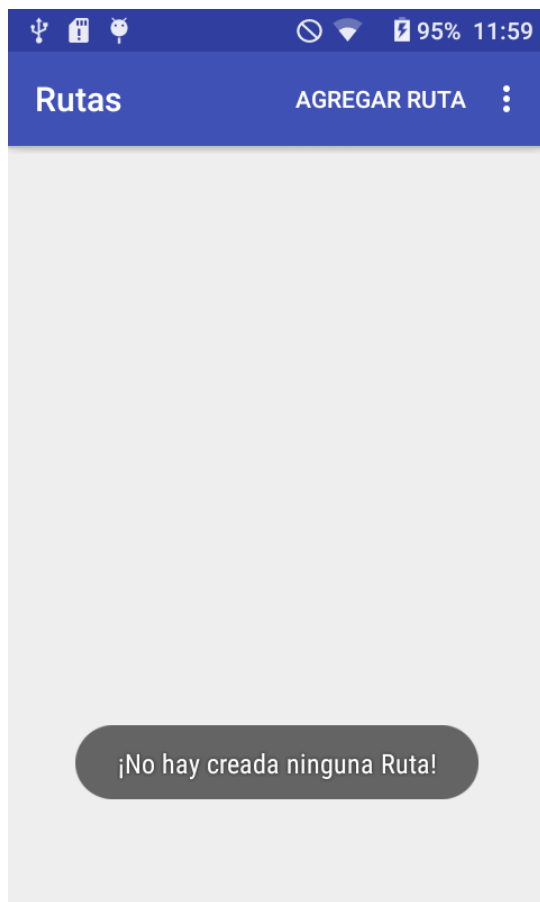
**Figura 51:** Editando Lugar

En los dos casos se permite aceptar (confirma los cambios y actualiza/agrega el lugar) y cancelar, si se pulsa aparece este mensaje:

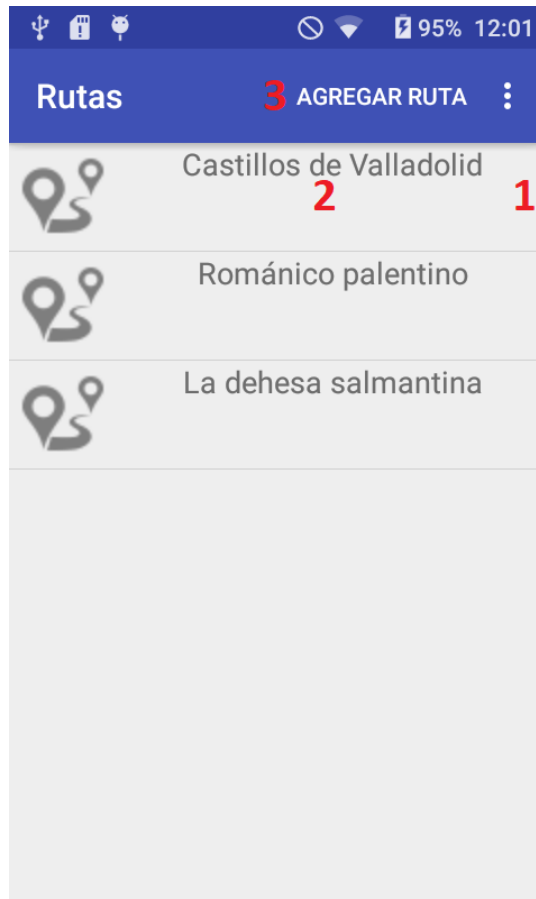
Si no se introduce el nombre aparece un mensaje de error:

## C.9. Lista Rutas

Esta pantalla muestra todas las rutas que se encuentran en la base de datos. En un primer momento, si no hay rutas, la pantalla muestra el siguiente aspecto:



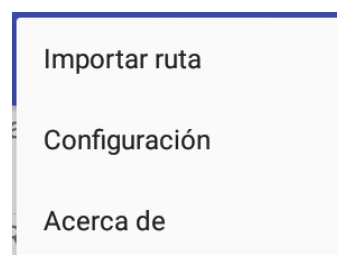
A medida que se crean rutas, se muestra así:



En esta pantalla se encuentran las siguientes opciones:

1. Elemento de la lista: hay tantos como rutas haya en la aplicación. Si se pulsa se muestran los lugares que tenga la ruta en Lista Lugares, si no tiene se informa.
2. Nombre de la ruta.
3. Agregar ruta: permite agregar rutas, abriendo Edición de nombre de Ruta. Los nombres de las rutas deben ser únicos.

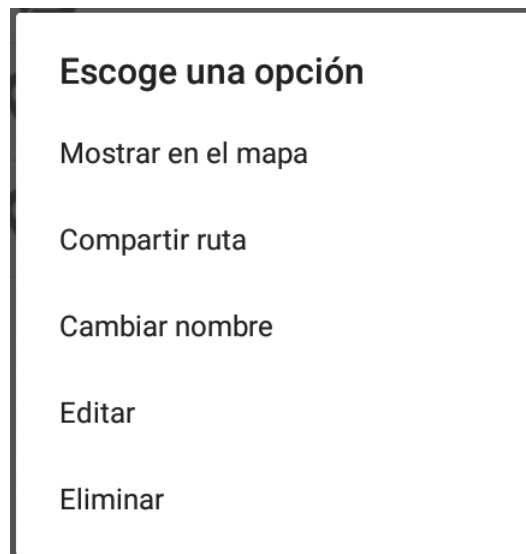
El menú es:



Las opciones son:

- Importar ruta: abre un selector de ficheros elegido por el usuario, y si se escoge un fichero que contiene una ruta, se agrega a la base de datos, incluyendo los lugares personalizados que pudiera tener. Los ficheros han de ser los generados por la aplicación, de tipo «.intour»
- Configuración: abre la pantalla de Preferencias
- Acerca de: abre la pantalla de Acerca de

Si se mantiene pulsado sobre un elemento de la lista se despliega un menú que permite varias opciones sobre esa ruta:



### C.9.1. Mostrar en el mapa

Se dibuja la ruta en el mapa. Si no se encuentra se informa. Si se tiene activada la localización se añade a la ruta. La ruta se busca en el orden en que aparecen los elementos.

### C.9.2. Compartir ruta

Se genera un fichero con los lugares que contiene la ruta, para poder compartirlo. Se deja a elección del usuario el medio que quiera para compartirlo. Los archivos generados son de tipo «.intour»

*Nota: actualmente existe un error al seleccionar Google Drive para compartirlo, ya que no se guarda correctamente el archivo, por lo que se desaconseja elegir esa opción.*

### C.9.3. Cambiar Nombre

Cambia el nombre de una ruta, se abre la pantalla de Edición de nombre de Ruta. Los nombres de las rutas deben ser únicos.

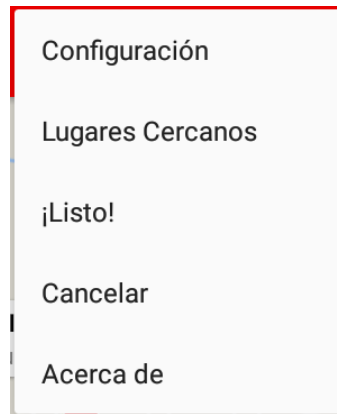
### C.9.4. Editar

Permite agregar lugares a una ruta. Al pulsarse se activa el modo de edición, que no permite trazar rutas o editar/eliminar lugares o ver las rutas, pero sí agregar lugares a la ruta. Este cambia el color de la aplicación, pasando de azul a rojo.

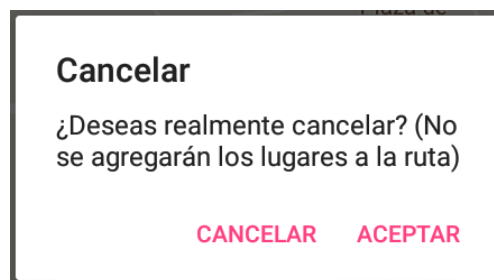


Figura 52: Modo de Edición de Ruta activo

Se modifican las opciones del menú de Mapa:

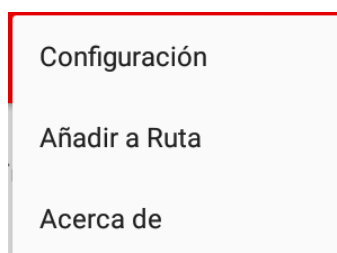


- ¡Listo!: confirma los cambios introducidos y finaliza el modo de edición.
- Cancelar: descarta los cambios, requiere confirmación.



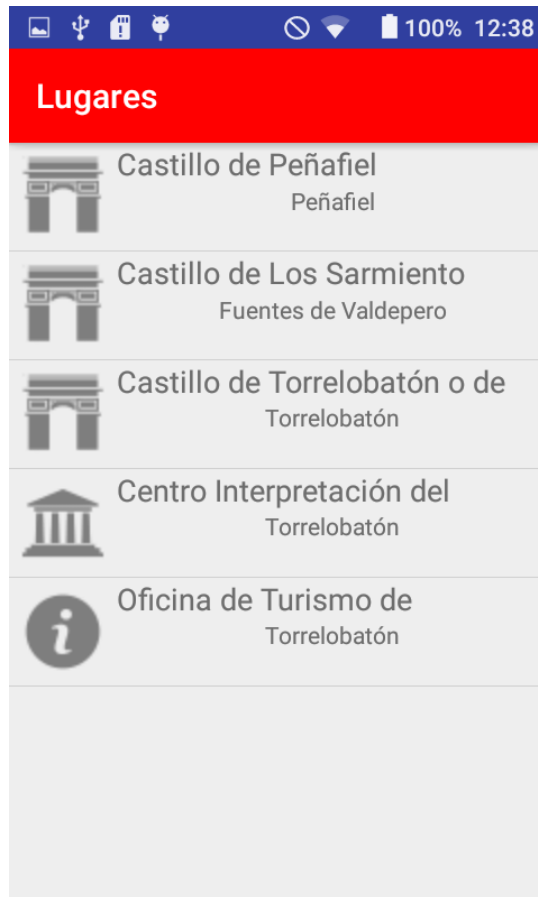
**Figura 53:** Diálogo mostrado al pulsar sobre cancelar

Se elimina la opción de trazar ruta del menú de Información de Lugar, y se añade la de «Añadir a ruta».

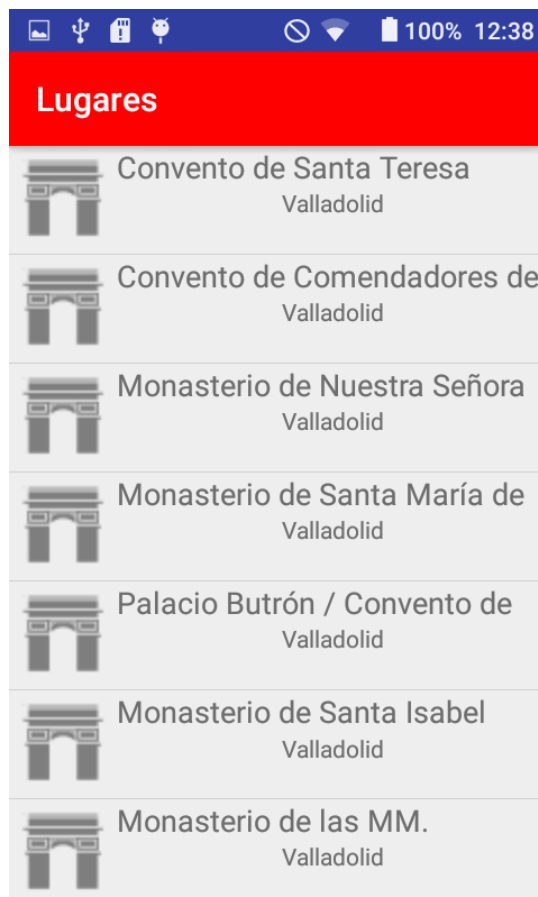


Se puede seguir buscando coincidencias y lugares cercanos y modificar las preferencias.





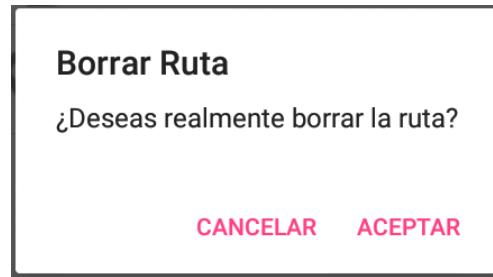
**Figura 54:** Modo de Edición de Ruta activo en búsqueda de coincidencias



**Figura 55:** Modo de Edición de Ruta activo en Lugares Cercanos

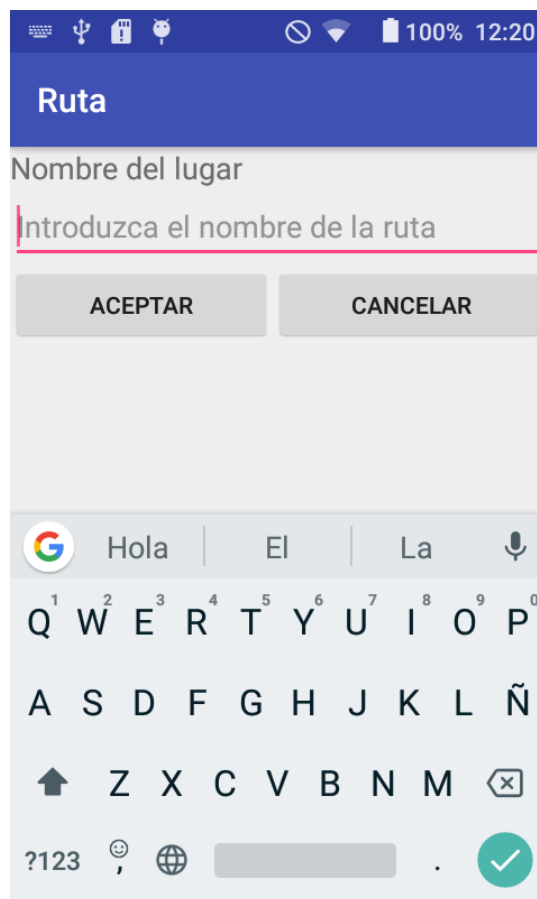
### C.9.5. Eliminar

Borra una ruta, requiere confirmación.

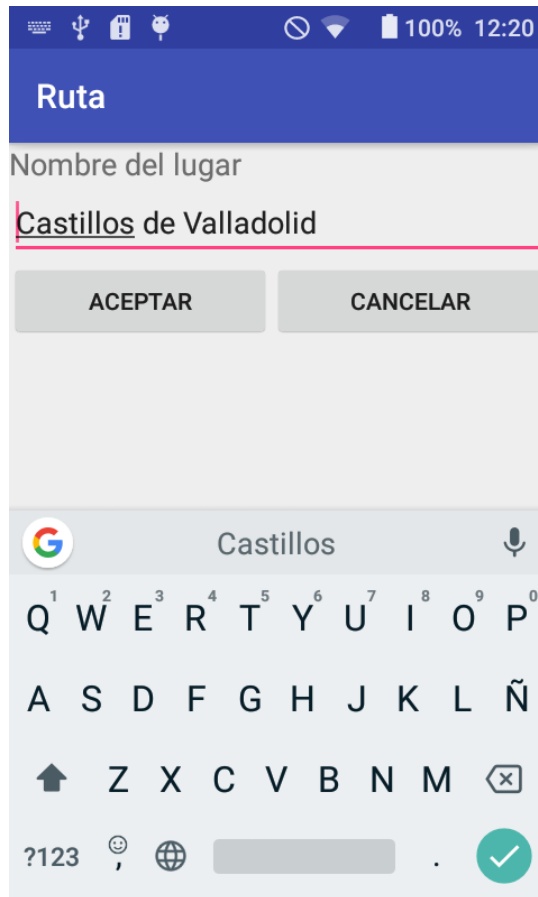


### C.10. Edición de nombre de Ruta

Permite especificar el nombre al agregar/editar el nombre de una ruta. Si se agrega este campo aparece vacío, si se edita aparece el nombre. El campo de nombre aparece subrayado.

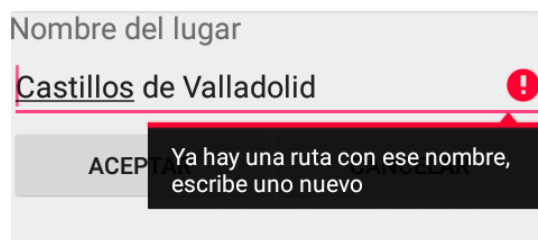


**Figura 56:** Agregar Ruta



**Figura 57:** Editar nombre de ruta

En ambos casos se puede aceptar y confirmar los cambios o cancelar y descartar los cambios. Si se introduce un nombre de ruta ya existente en la aplicación, se muestra un mensaje.



## C.11. Acerca de

Muestra información sobre la aplicación, si se pulsa sobre los enlaces se puede acceder a las direcciones web y escribir un correo.



**Figura 58:** Pantalla Acerca De

## D. Diagramas de acceso a datos

En esta sección se incluyen todos los diagramas de secuencia de acceso a datos que han quedado indicados a lo largo de la memoria.

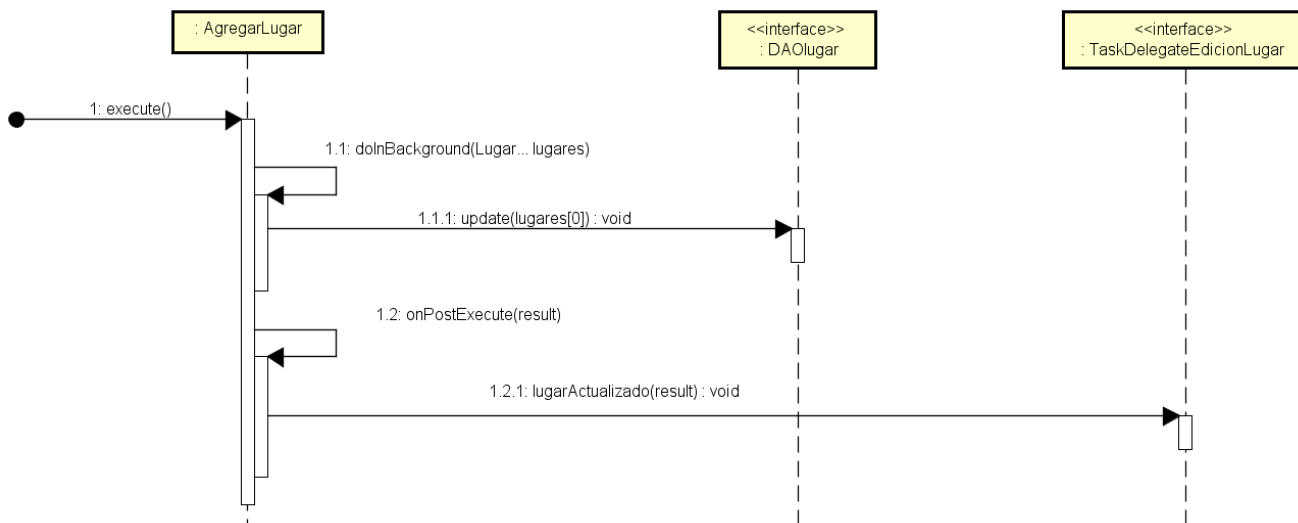


Figura 59: Actualizar Lugar

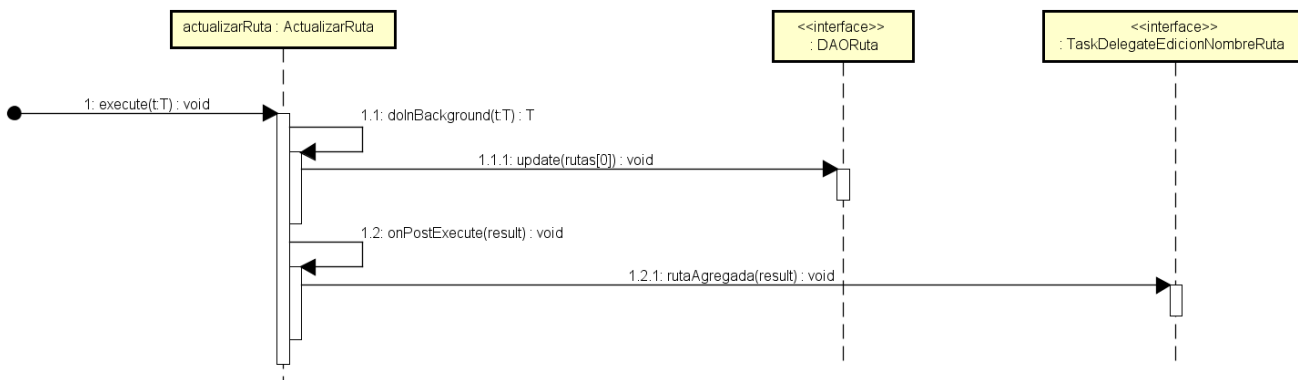


Figura 60: Actualizar Ruta

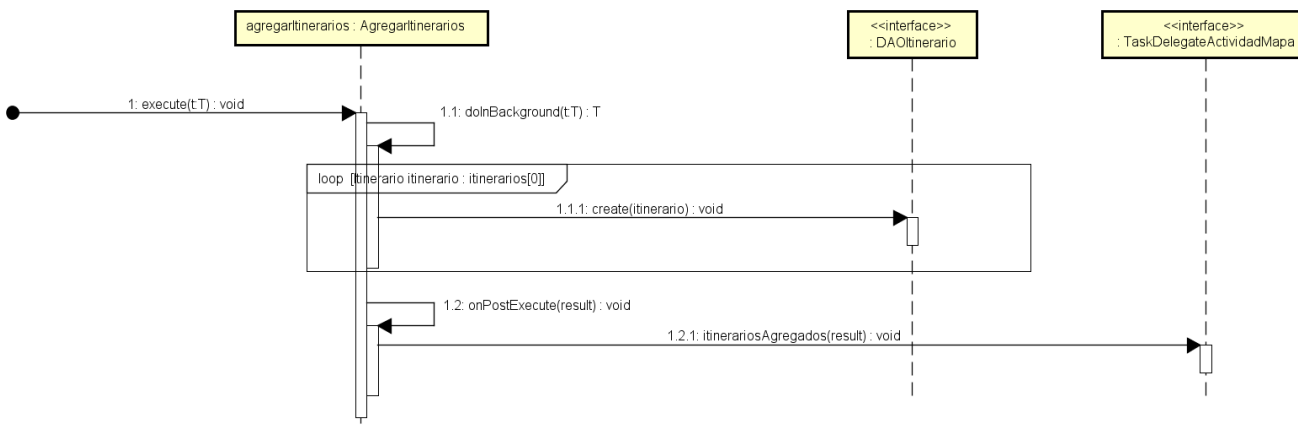


Figura 61: Agregar Itinerarios

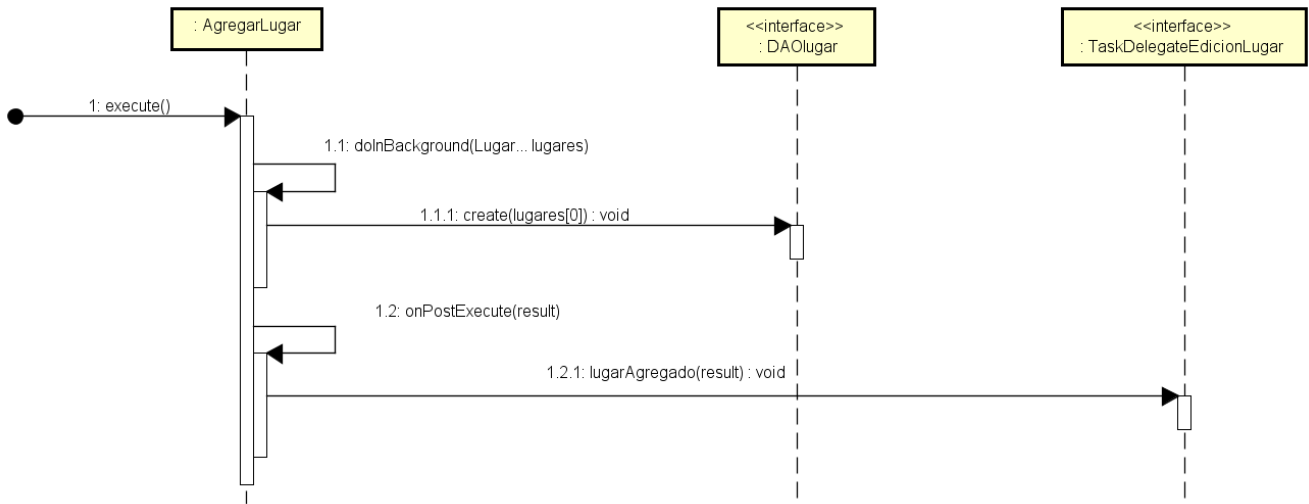


Figura 62: Agregar Lugar

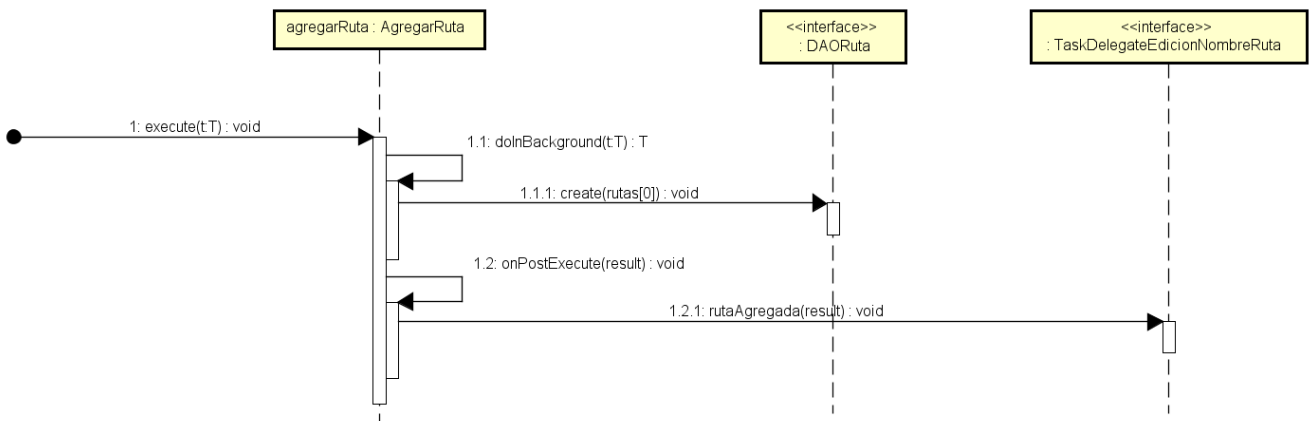


Figura 63: Añadir Ruta

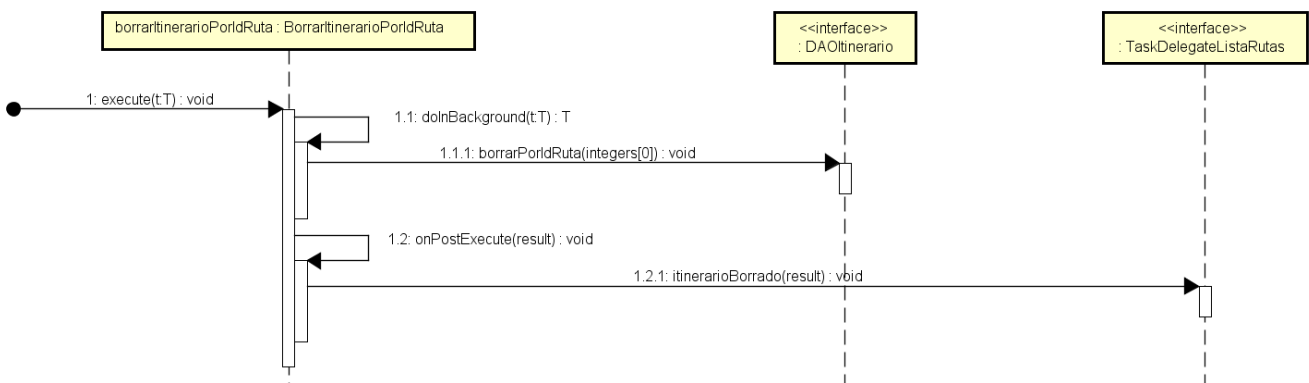


Figura 64: Borrar Itinerario por Id Ruta

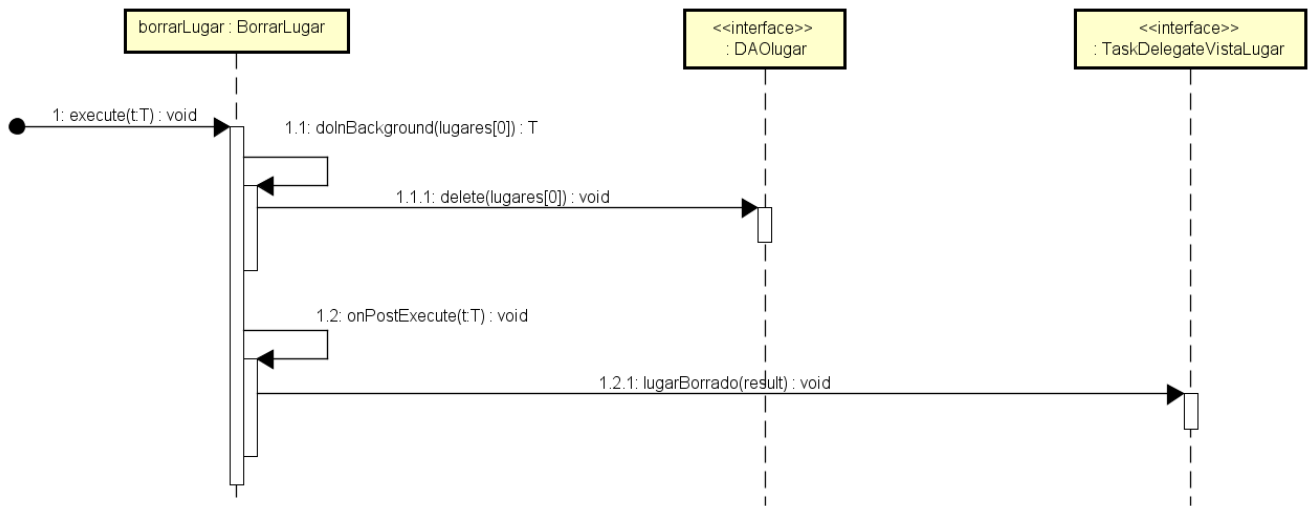


Figura 65: Borrar Lugar

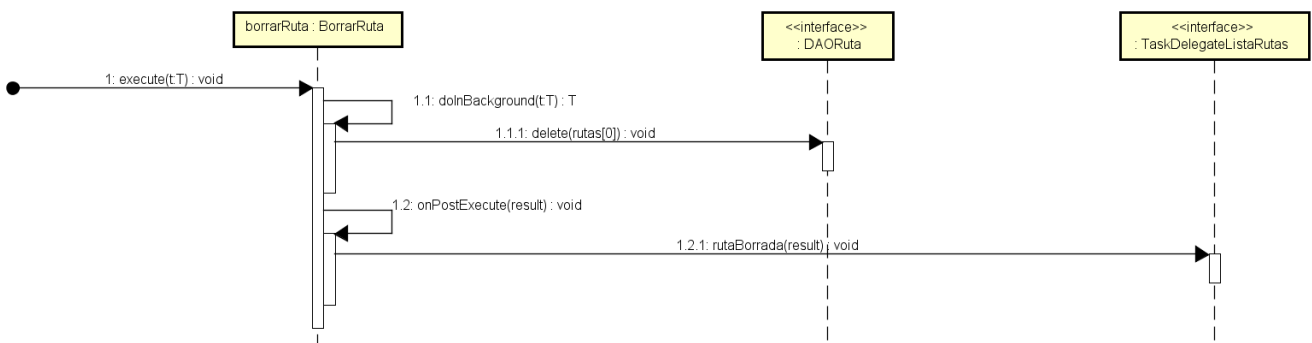


Figura 66: Borrar Ruta

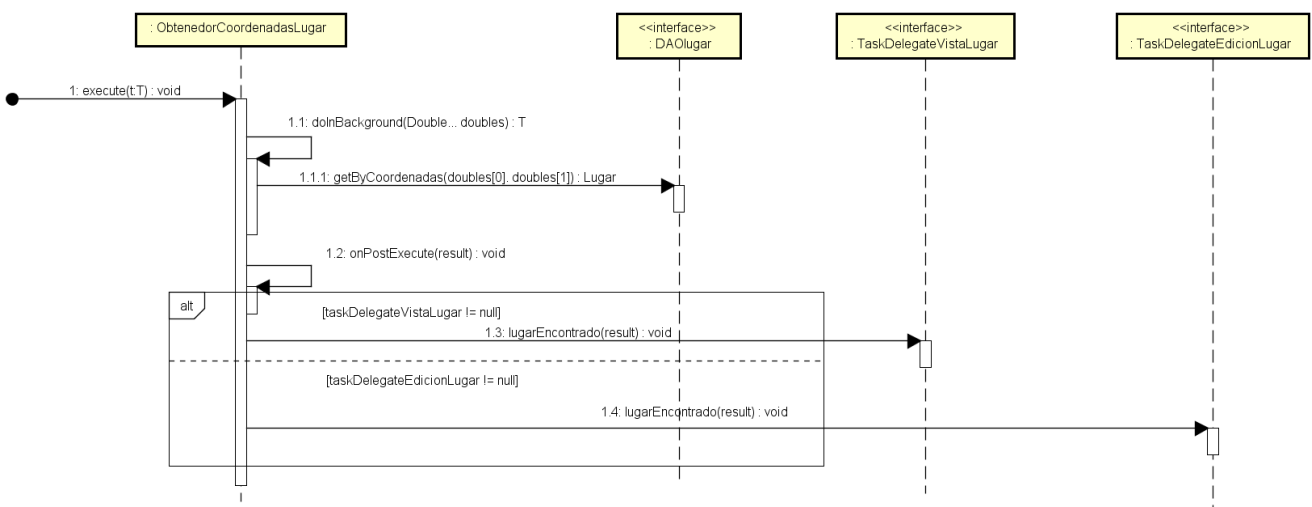


Figura 67: Obtener Coordenadas Lugar

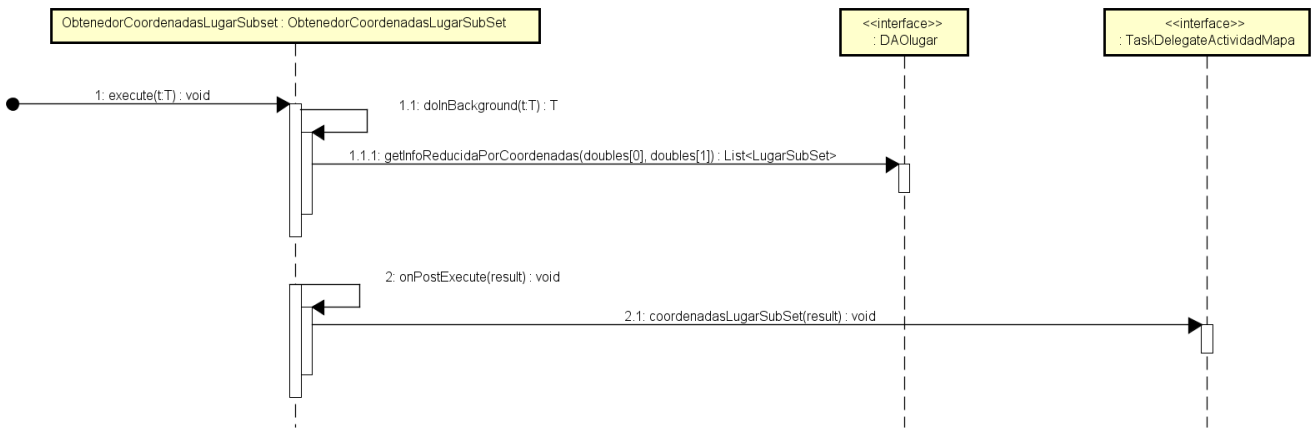


Figura 68: Obtener coordenadas LugarSubSet

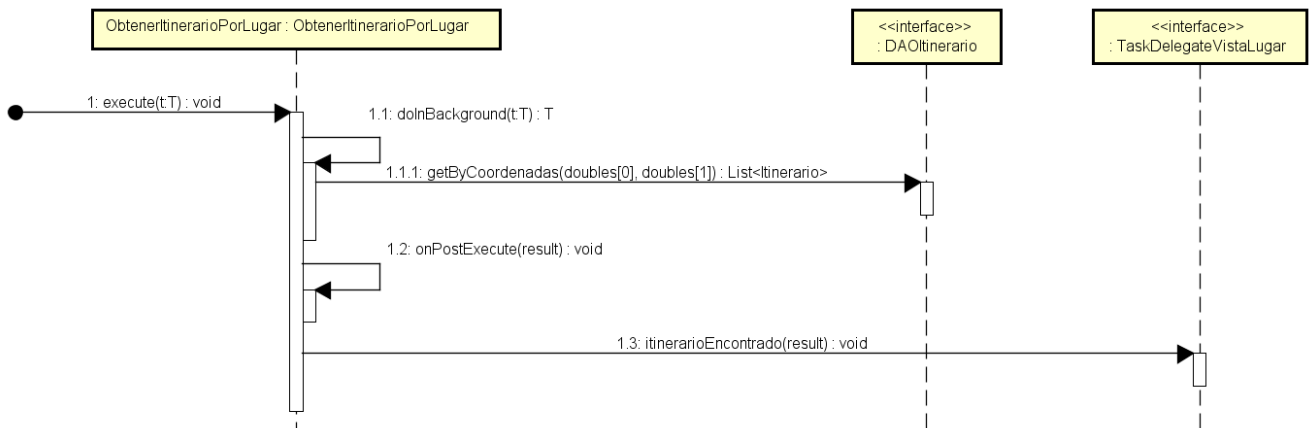


Figura 69: Obtener Itinerario por Lugar

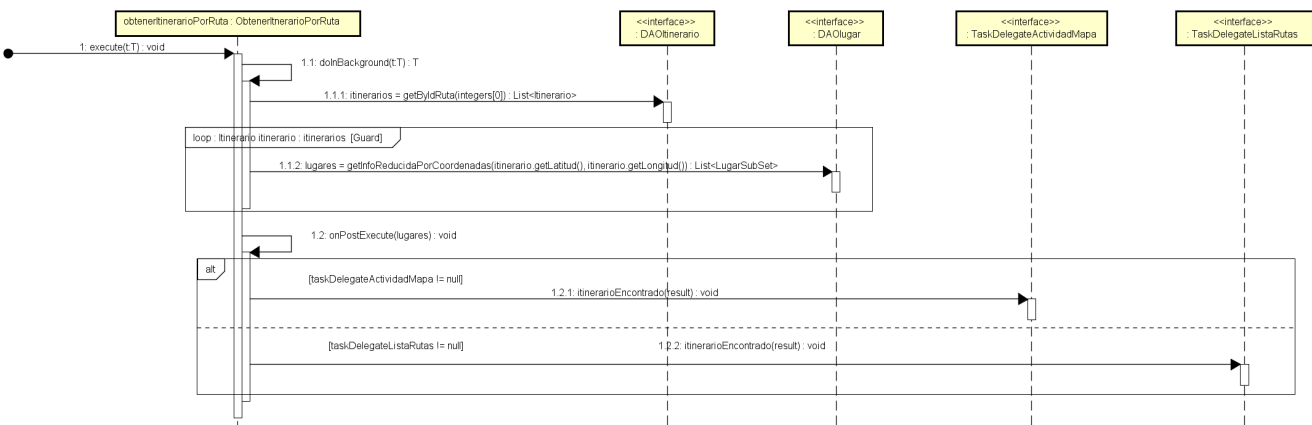


Figura 70: Obtener Itinerario por Ruta



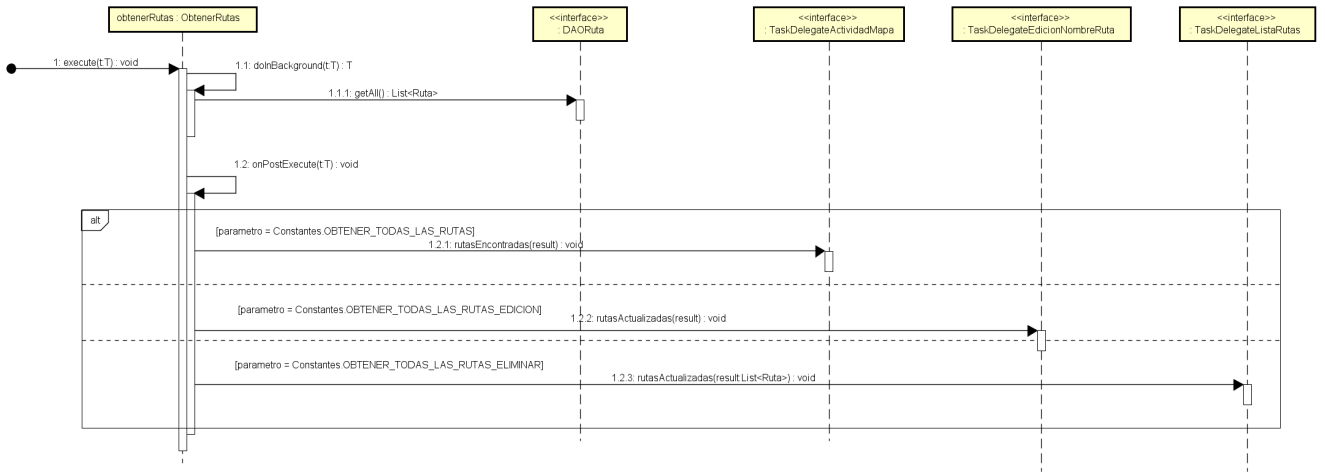


Figura 71: Obtener Rutas

