



Universidad
Zaragoza

Ingeniería del Software

Práctica 5: Patrones de diseño e implementación

Dorian Boleslaw Wozniak (817570@unizar.es)

Sergio Martinez Lahoz (801621@unizar.es)

En Zaragoza, a 19 de enero de 2022

Resumen

Las prácticas asociadas a este proyecto consisten en análisis, diseño, implementación y pruebas de una interfaz para una aplicación para dispositivos Android que permite gestionar las habitaciones de un hotel rural, y además gestionar las reservas realizadas por clientes asociadas a estas habitaciones.

En la fase de análisis, una vez recogidos los requisitos a cumplir y los casos de uso que se darán al utilizar la aplicación, se ha diseñado un modelo estático y dinámico inicial mediante diagramas de clases y de secuencia, que describen las componentes y el comportamiento de los elementos que forman parte del problema.

En la fase de diseño se ha concretado la aproximación a tomar para implementarlo mediante el entorno de desarrollo de Android. Se ha obtenido un modelo para el almacenamiento de la información, los subsistemas y componentes que formarán la aplicación y un modelo estático y dinámico más elaborado y que se asemeja a lo que se necesita a la hora de implementarlo.

Nota: Esta memoria es un documento en desarrollo y será actualizado durante las prácticas posteriores.

Índice

Resumen	2
Índice	3
1. Introducción y objetivos	4
2. Requisitos	5
2.1. Catálogos de requisitos	5
2.2. Diagrama de casos de uso	7
2.3. Descripciones de casos de uso	8
3. Análisis	13
3.1. Modelo estático	13
3.2. Modelo dinámico	14
4. Prototipado del usuario.	23
4.1. Prototipos de pantallas.	23
4.2. Mapa de navegación del sistema	27
5. Diseño del sistema y diseño de objetos	29
5.1. Modelo lógico de la base de datos	30
5.2. Diagrama de clases a nivel de diseño	32
5.3. Diagrama de paquetes	33
5.4. Diagrama de componentes	33
5.5. Diagrama de despliegue	34
5.6. Diagramas de secuencia a nivel de diseño	35
Anexo: Horas dedicadas	46

1. Introducción y objetivos

El objetivo del proyecto es crear una interfaz de una aplicación móvil para dispositivos Android. La aplicación está destinada a un propietario o propietaria de un hotel rural que quiera facilitar su gestión, cuyas reservas son comunicadas por los clientes de forma informal.

Para elaborar las prácticas, se va a cumplir una serie de objetivos:

- Desarrollar un **catálogo de requisitos** a partir del enunciado.
- Obtener un **diagrama de casos de uso** a partir de los requisitos, y **describir su flujo** de forma textual.
- Realizar el **modelado estático** mediante un **diagrama de clase**.
- Realizar el **modelado dinámico** de cada caso de uso mediante **diagramas de secuencia**, que utilizarán las clases descritas anteriormente.
- Proponer un **prototipo** inicial de la interfaz a desarrollar.
- Definir el **modelo de la base de datos** a utilizar para el almacenamiento persistente.
- Revisar los **diagramas de clase y de secuencia** a nivel de diseño, incluyendo las interacciones con el entorno de desarrollo de Android/
- Añadir los **diagramas de paquetes, componentes y despliegue** para definir la organización de los subsistemas de la aplicación y cómo interactúa con otros sistemas en el dispositivo.
- Revisar apartados anteriores de forma iterativa conforme se detecten errores, carencias o formas mejores de solucionar problemas.

2. Requisitos

2.1. Catálogos de requisitos

A partir del enunciado descrito anteriormente, se ha definido los requisitos funcionales y no funcionales en una **tabla de requisitos**:

Código	Descripción
F1	Una habitación consta de un identificador único , una descripción , un número máximo de ocupantes , un precio por cada noche para un solo ocupante, y un porcentaje de recargo adicional que se aplica al precio base por cada inquilino extra
F2	El sistema debe permitir añadir habitaciones nuevas
F3	El sistema debe permitir consultar habitaciones creadas
F4	Las consultas de habitaciones se deben poder ordenar por identificador , número máximo de ocupantes o precio por noche
F5	El sistema debe permitir modificar habitaciones previamente creadas
F6	El sistema debe permitir eliminar habitaciones previamente creadas
F7	Una reserva consta de un nombre del cliente , el número móvil del cliente, la fecha de entrada , la fecha de salida , y una selección de habitaciones reservadas junto al número de ocupantes en cada habitación seleccionada
F8	El sistema debe permitir añadir reservas nuevas
F9	El sistema debe permitir consultar reservas creadas
F10	Las consultas de reservas se deben poder ordenar por nombre del cliente , número de móvil o fecha de entrada
F11	El sistema debe permitir modificar reservas previamente creadas
F12	El sistema debe permitir eliminar reservas previamente creadas
F13	El sistema debe permitir calcular el precio total de una reserva a partir de los ocupantes por cada habitación seleccionada en la reserva
F14	El sistema debe asegurarse que una reserva sea válida

F14.1	Una reserva se considera válida si no se producen solapes con otras reservas , es decir, no existe otra reserva para una habitación seleccionada con la fecha de entrada y/o salida posterior a la fecha de entrada a la reserva a validar; y que no se supere la cantidad máxima de ocupantes por habitación seleccionada
F15	El sistema debe permitir enviar un mensaje al teléfono móvil del cliente de una reserva con los datos de esta.
NF1	El mensaje de texto para una reserva confirmada contiene la información de la reserva junto al precio total de la reserva
NF2	El sistema deberá poder almacenar unas 100 habitaciones
NF3	El sistema poder almacenar unas 2000 reservas

Fig. 1: Tabla de requisitos funcionales y no funcionales obtenidos a partir del análisis del enunciado dado

Adicionalmente, se han definido las siguientes **restricciones** sobre el sistema:

- El sistema será desarrollado para dispositivos Android, siendo retrocompatible hasta dispositivos con la versión de Android 4.1 (*Jellyfish*).

2.2. Diagrama de casos de uso

En base a los requisitos obtenidos anteriormente, se ha obtenido el siguiente **diagrama de casos de uso**:

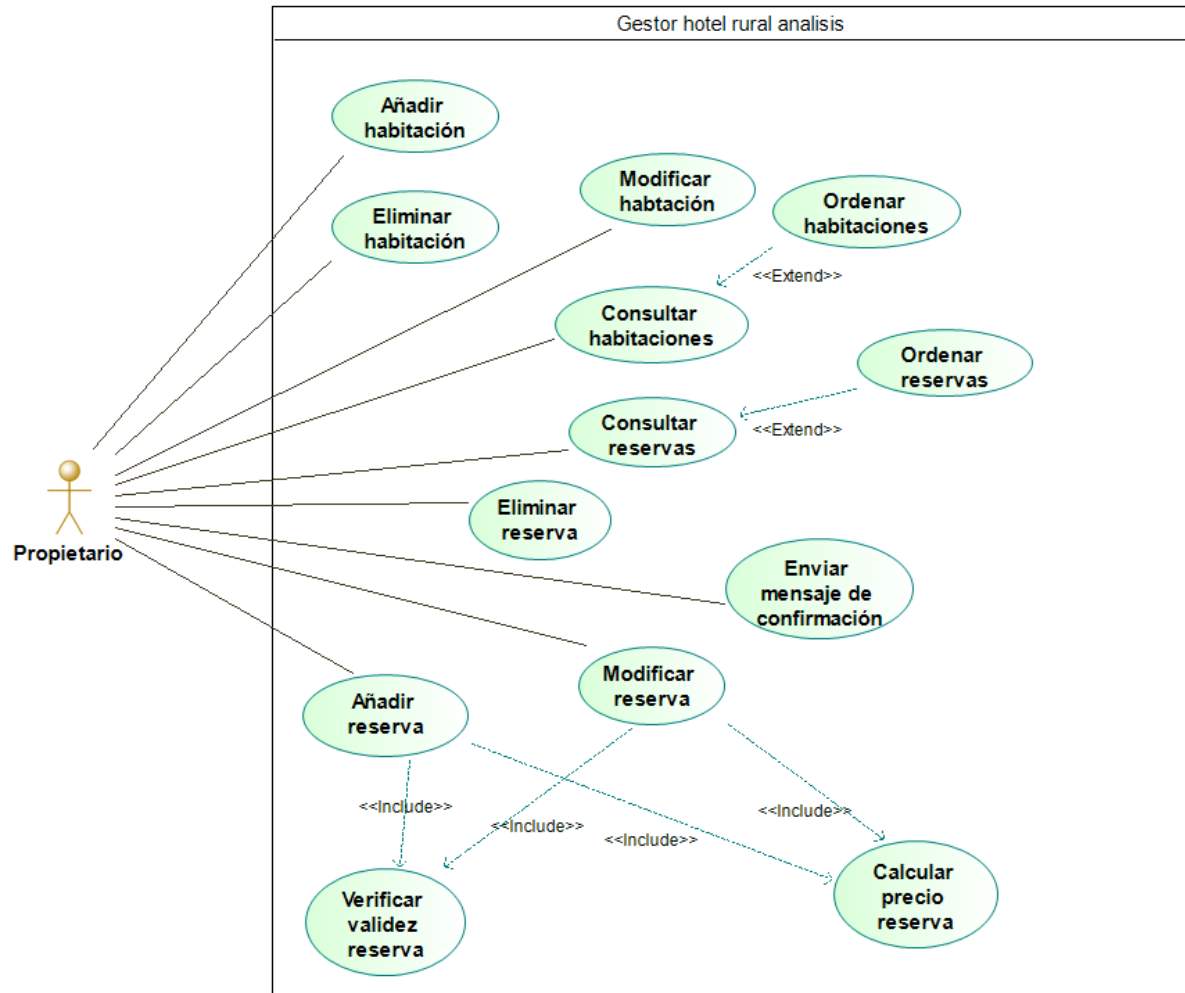


Fig. 2: Diagrama de casos de uso obtenido a partir de los requisitos funcionales definidos

Cabe destacar que el sistema, siguiendo con lo expuesto por el enunciado, solo existe un actor: el propietario del hotel. En vez de ser el cliente quien añada una reserva a través de la aplicación, el propietario añadirá las reservas conforme le sean comunicadas mediante otros canales de comunicación.

2.3. Descripciones de casos de uso

Ahora vamos a realizar las descripciones de los casos de uso:

Caso de uso: Añadir habitación

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de añadir habitación.
- El propietario tiene que introducir un identificador, una descripción de la habitación, un número máximo de ocupantes que puede tener la ocupación, un precio por noche para un solo ocupante y un porcentaje de recargo aplicado por cada ocupante extra.
- El sistema guarda la habitación creada y el caso de uso acaba.

Caso de uso: Eliminar habitación

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de eliminar una habitación.
- El sistema muestra a continuación al usuario una selección de habitaciones ya creadas que se pueden eliminar.
- El propietario selecciona la habitación que desea eliminar.
- El sistema elimina la habitación seleccionada.
- El caso de uso acaba cuando el sistema notifica que la habitación ha sido eliminada correctamente al propietario.

Flujo de eventos alternativo:

- Si no hay ninguna habitación creada en el sistema, se notificará al propietario que no hay habitaciones disponibles para eliminar.

Caso de uso: Modificar habitación

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de modificar una habitación.
- El sistema muestra a continuación al usuario una selección de habitaciones ya creadas que se pueden modificar.
- El propietario selecciona la habitación que desea modificar.
- El sistema le muestra los datos de la habitación escogida.
- El propietario modifica los datos que el necesite.
- El caso de uso acaba cuando el sistema notifica que la habitación ha sido modificada por el propietario.

Flujo de eventos alternativo:

- Si no hay ninguna habitación creada en el sistema, se notifica al propietario que no hay habitaciones disponibles para modificar.

Caso de uso: Consultar habitaciones

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de consultar habitaciones.
- El sistema le muestra las habitaciones que hay disponibles.
- El usuario selecciona una de las habitaciones para consultarla
- El sistema muestra la información disponible sobre la habitación
- El caso de uso acaba cuando el usuario sale de la opción para consultar habitaciones

Flujo de eventos alternativos:

- El sistema no muestra ninguna habitación si no hay habitaciones creadas previamente y le mostrará un mensaje de que no hay habitaciones creadas.

Flujo de eventos alternativos:

- El usuario puede seleccionar el orden en el que se muestran las habitaciones: por identificador (por defecto), por número máximo de ocupantes o por el precio por noche.
- El sistema vuelve a mostrar las habitaciones disponibles en el orden seleccionado

Caso de uso: Consultar reservas

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de consultar reservas.
- El sistema le muestra las reservas que hay disponibles.
- El usuario selecciona una de las reservas para consultarla
- El sistema muestra la información disponible sobre la reserva
- El caso de uso acaba cuando el usuario sale de la opción para consultar reservas

Flujo de eventos alternativos:

- El sistema no puede consultar reservas si no hay previamente reservas creadas.

Flujo de eventos alternativos:

- El usuario puede seleccionar el orden en el que se muestran las reservas: nombre de cliente (por defecto), número de móvil o fecha de entrada
- El sistema vuelve a mostrar las reservas disponibles en el orden seleccionado

Caso de uso: Añadir reserva

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de añadir reserva.
- El propietario tiene que introducir un nombre de cliente, su número de teléfono, las fechas de entrada y salida, una selección de habitaciones reservadas y el número de ocupantes por habitación.
- *Include*: El sistema verifica la validez de la reserva.
- *Include*: El sistema calcula el precio total de la reserva.
- El sistema guarda la reserva creada y el caso de uso acaba.

Flujo de eventos alternativo:

- Si el sistema determina que la reserva no es válida, el usuario deberá volver a introducir los datos.

Caso de uso: Eliminar reserva

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de eliminar una reserva.
- El sistema muestra a continuación al usuario una selección de reservas ya creadas que se pueden eliminar.
- El propietario selecciona la reserva que desea eliminar.
- El sistema elimina la reserva seleccionada.
- El caso de uso acaba cuando el sistema notifica que la reserva ha sido eliminada correctamente al propietario.

Flujo de eventos alternativo:

- Si no hay ninguna reserva creada en el sistema, se notificará al propietario que no hay reservas disponibles para eliminar.

Caso de uso: Modificar habitación

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de modificar una reserva.
- El sistema muestra a continuación al usuario una selección de reservas ya creadas que se pueden modificar.
- El propietario selecciona la reserva que desea modificar.
- El sistema le muestra los datos de la reserva escogida.
- El propietario modifica los datos que el necesite.
- *Include*: El sistema verifica la validez de la reserva.
- *Include*: El sistema calcula el precio total de la reserva.
- El caso de uso acaba cuando el sistema notifica que la habitación ha sido modificada por el propietario.

Flujo de eventos alternativo:

- Si no hay ninguna reserva creada en el sistema, se notifica al propietario que no hay reservas disponibles para modificar.

Flujo de eventos alternativo:

- Si el sistema determina que la reserva no es válida, el usuario deberá volver a introducir los datos.

Caso de uso: Enviar mensaje de confirmación de reserva

Flujo de eventos principal:

- El caso de uso comienza cuando el propietario selecciona la opción de enviar un mensaje de confirmación de una reserva.
- El sistema muestra a continuación al usuario una selección de reservas ya creadas/
- El propietario selecciona la reserva de la que desea enviar un mensaje.
- El sistema envía un mensaje al teléfono móvil del cliente para esa reserva cuyos contenidos son los datos de la reserva.
- El caso de uso acaba cuando el sistema informa que el mensaje se ha enviado.
- Flujo de eventos alternativo:
- Si no hay ninguna reserva creada en el sistema, se notifica al propietario que no hay reservas disponibles.

3. Análisis

3.1. Modelo estático

A continuación, se ha definido el modelo estático de la aplicación a realizar. El modelo estático se ha plasmado mediante el uso de un **diagrama de clases**:

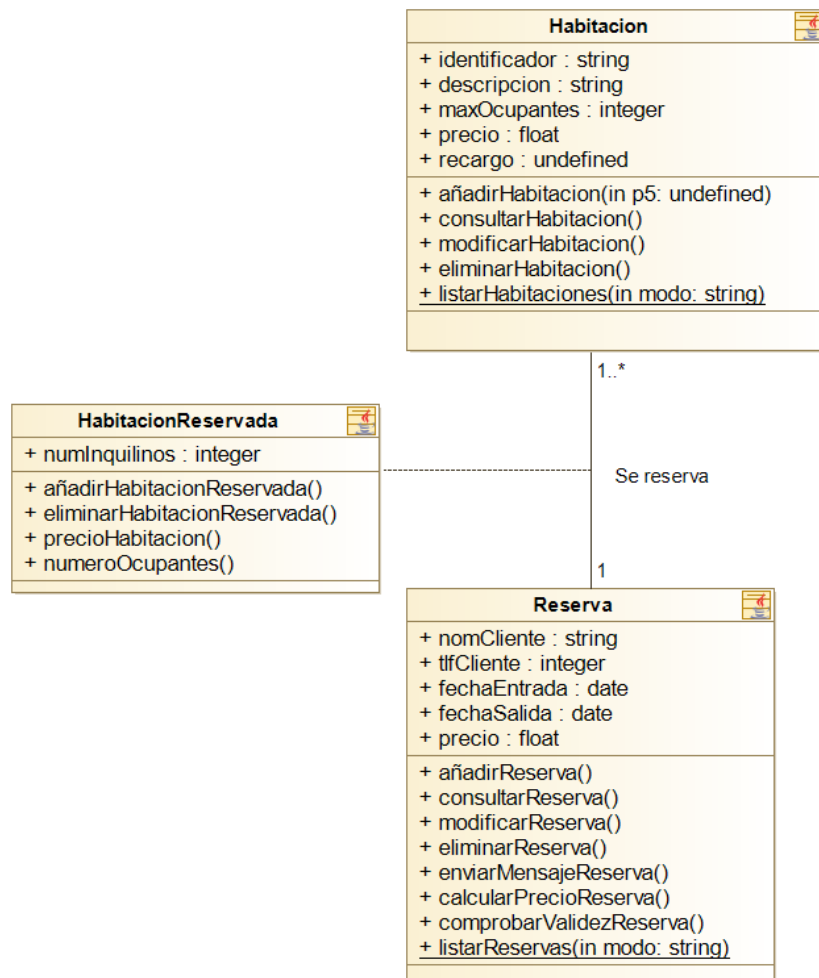


Fig. 3: Diagrama de clases de la aplicación propuesta

Se han creado tres clases: **Habitación**, **Reserva**, y **HabitaciónReservada**. Las dos primeras clases almacenan sus respectivos datos, junto a métodos para crearlas, eliminarlas, consultarlas, modificarlas y obtener una lista de estas. Además, la reserva contiene un método para enviar un mensaje de confirmación, y métodos privados que comprueban la validez de estas al añadir una nueva reserva o modificarla, y para calcular su precio. Una reserva debe estar relacionada con al menos una habitación tal que sea ocupada por un número de personas, lo cual se representa mediante la última clase.

3.2. Modelo dinámico

Caso de uso: Añadir habitación

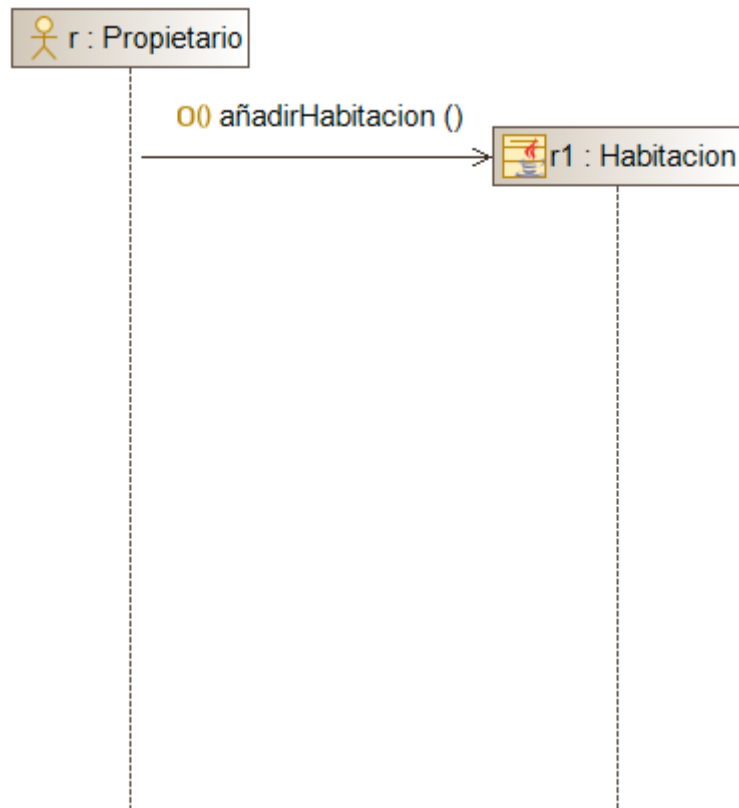


Fig. 4: Diagrama de secuencia del caso de uso “Añadir habitación”

Caso de uso: Eliminar habitación

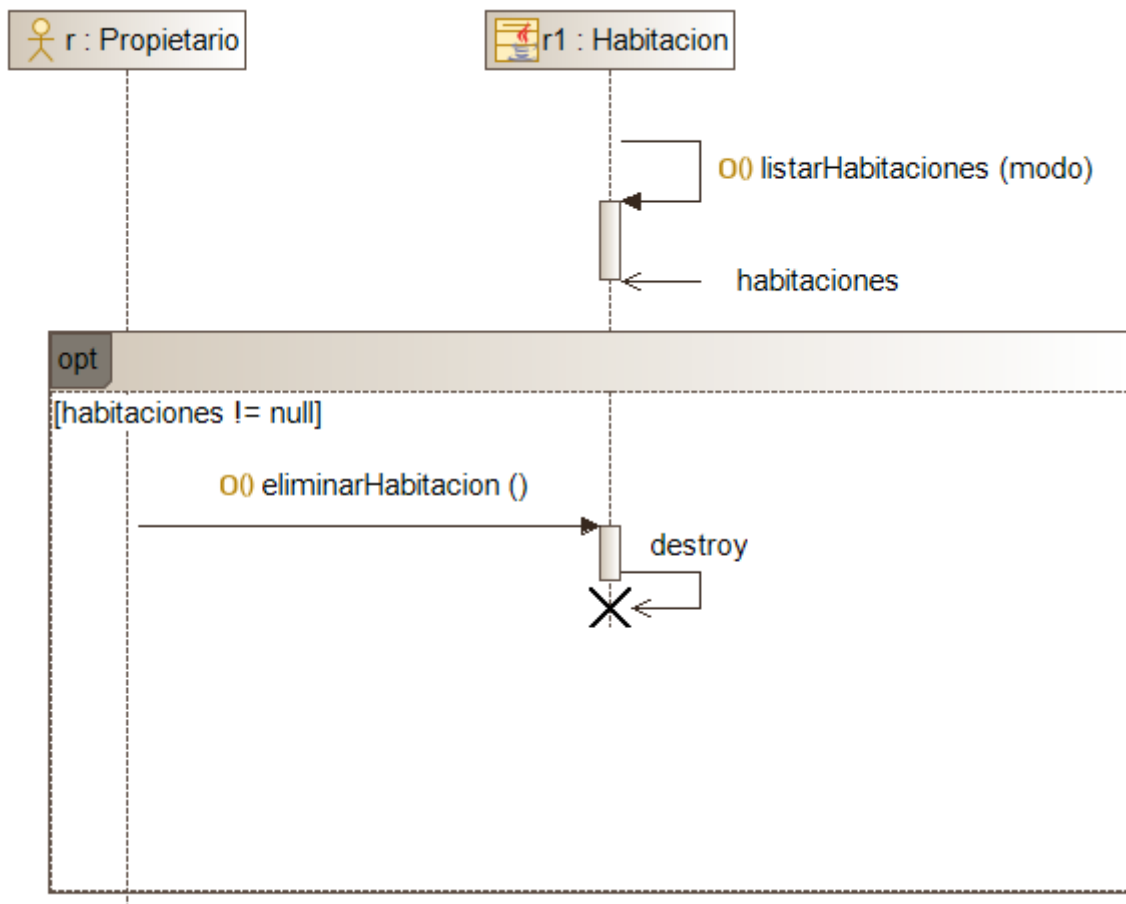


Fig. 5: Diagrama de secuencia del caso de uso “Eliminar habitación”

Caso de uso: Modificar habitación

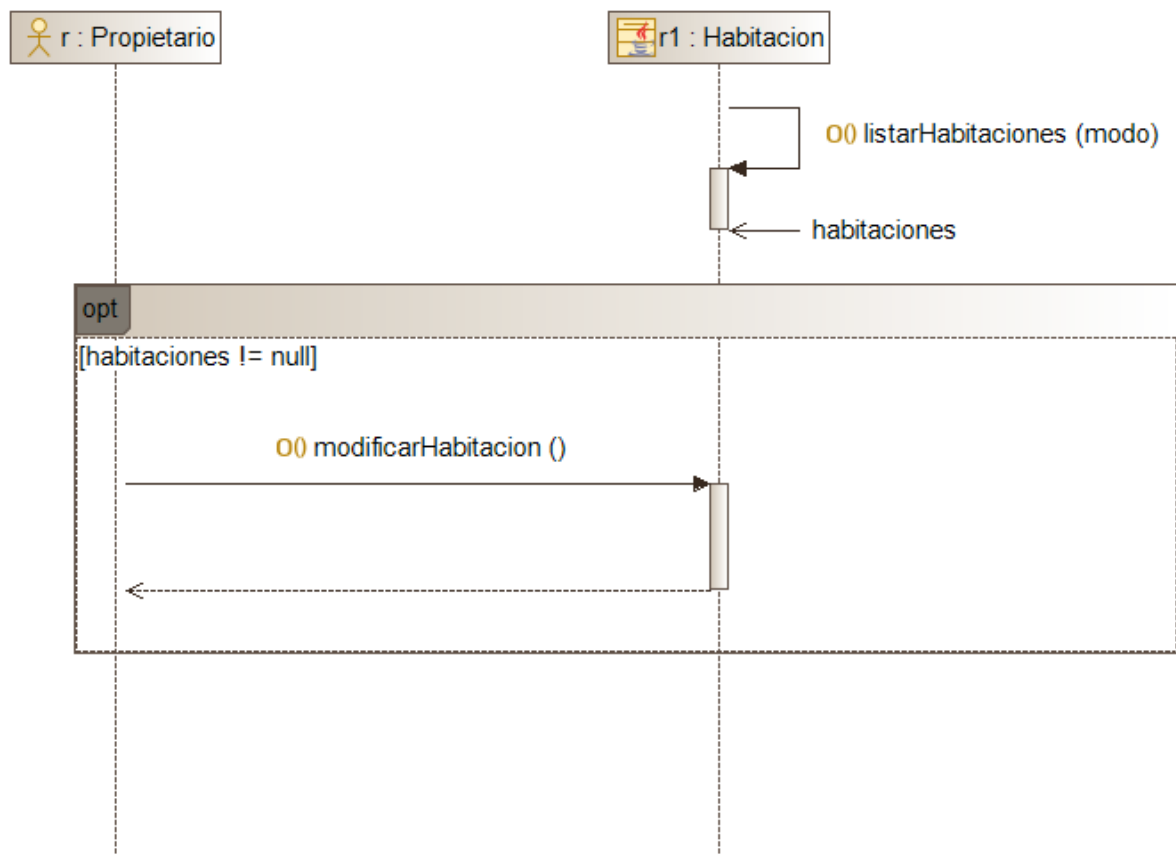


Fig. 6: Diagrama de secuencia del caso de uso “Modificar habitación”

Caso de uso: Consultar habitaciones

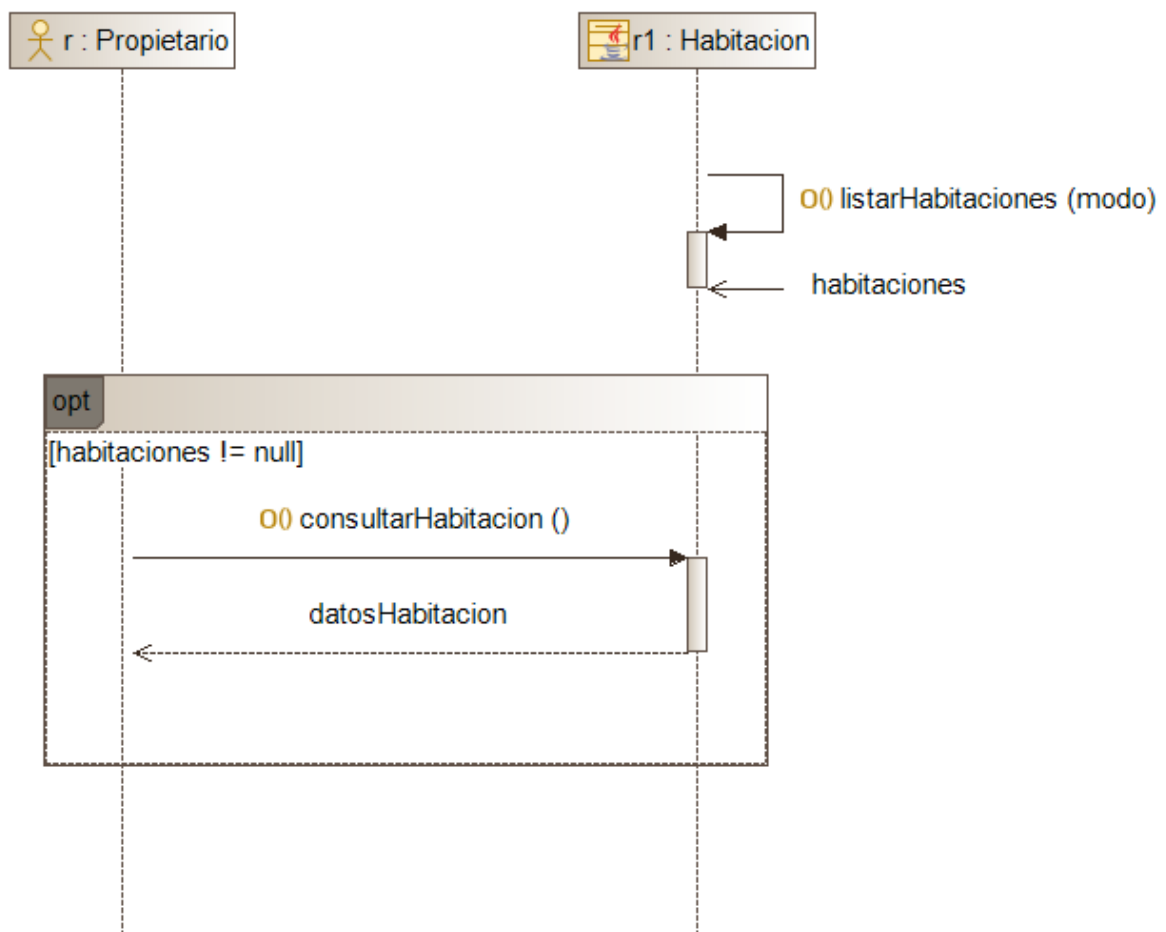


Fig. 7: Diagrama de secuencia del caso de uso "Consultar habitaciones"

Caso de uso: Añadir reserva

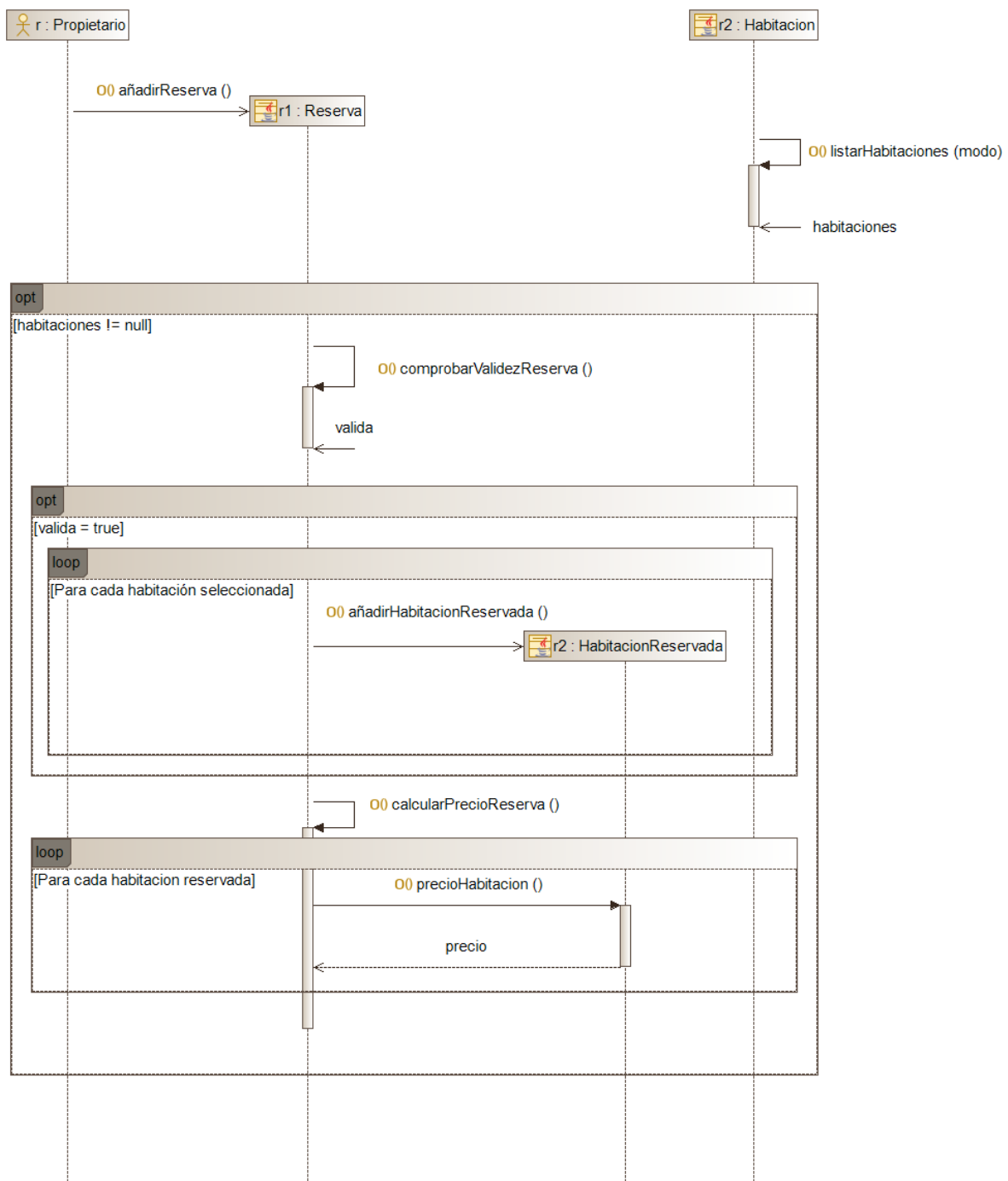


Fig. 8: Diagrama de secuencia del caso de uso "Añadir reserva" comprobar validez después de añadirHabitacionReservada

Caso de uso: Eliminar reserva

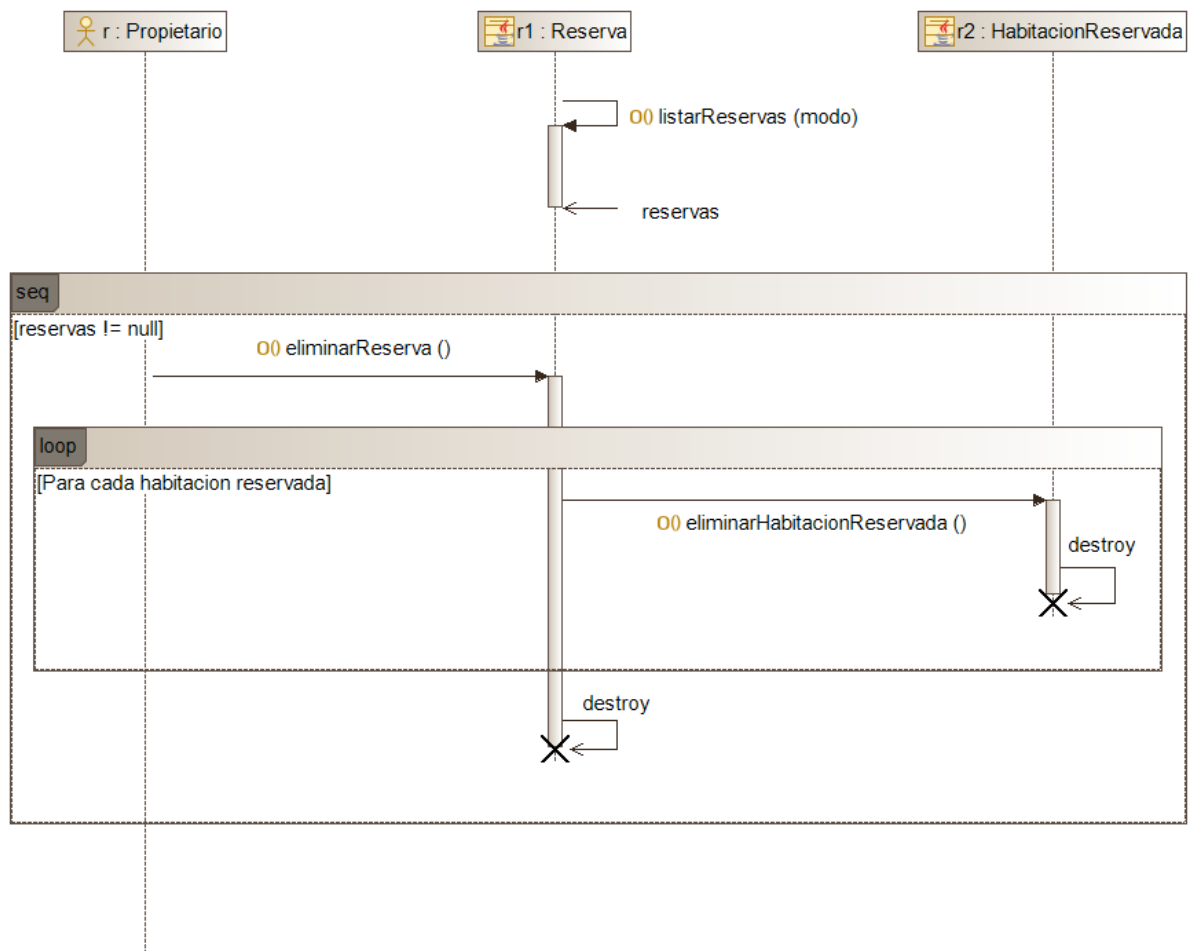


Fig. 9: Diagrama de secuencia 2 del caso de uso "Eliminar reserva"

Caso de uso: Modificar reserva

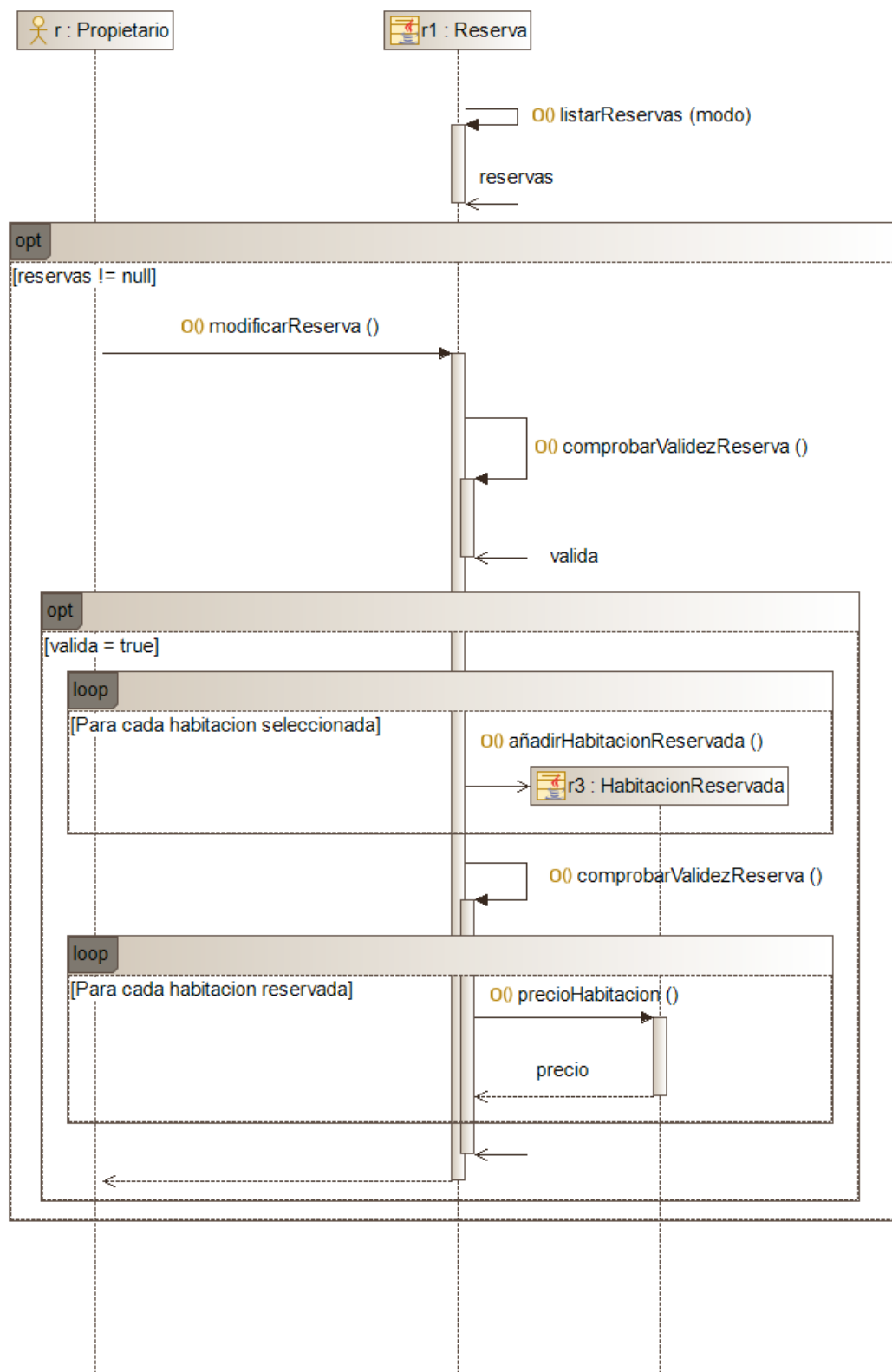


Fig. 10: Diagrama de secuencia del caso de uso "Modificar reserva"

Caso de uso: Consultar reservas

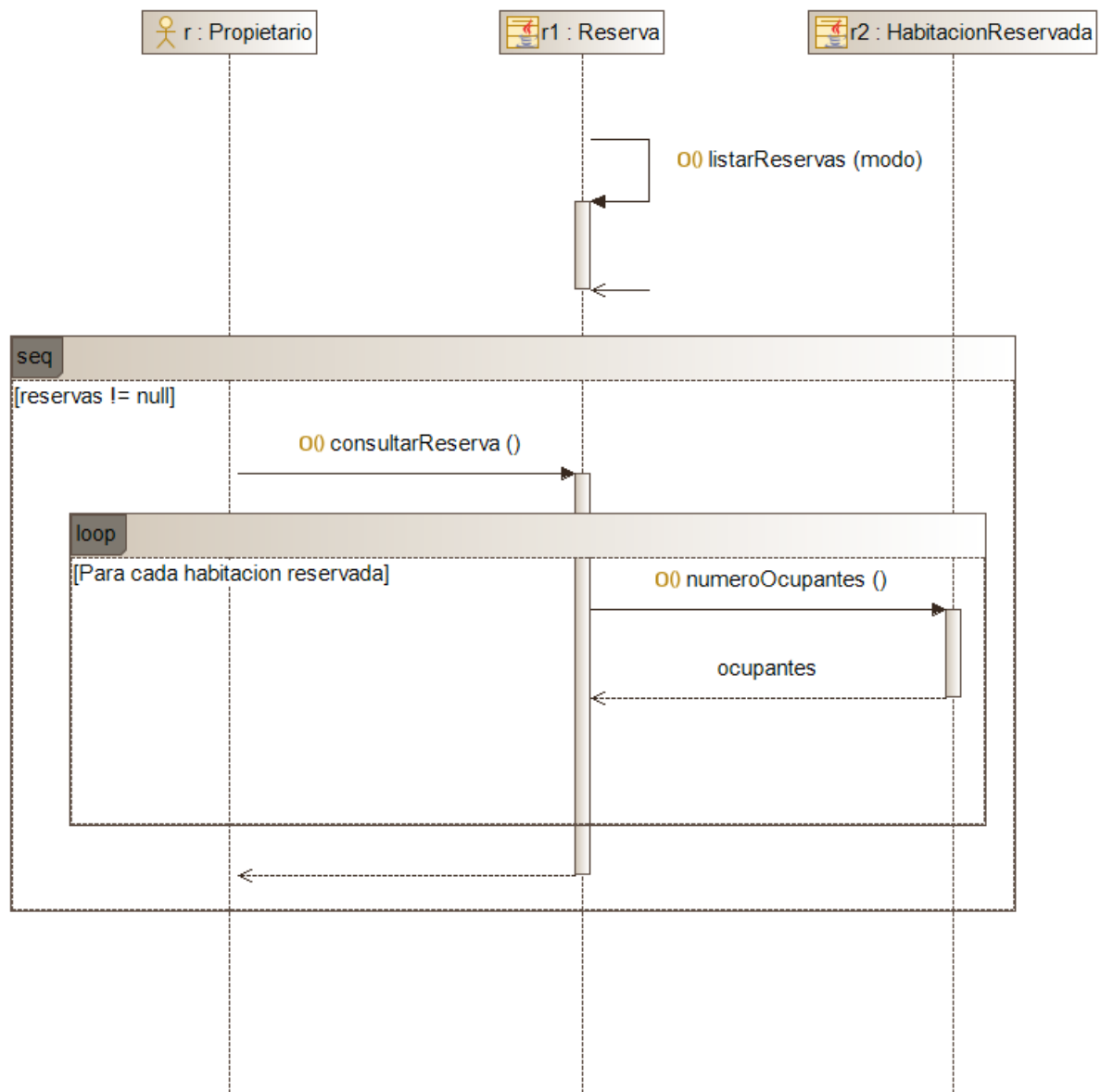


Fig. 11: Diagrama de secuencia del caso de uso "Consultar reservas"

Caso de uso: Enviar mensaje de confirmación

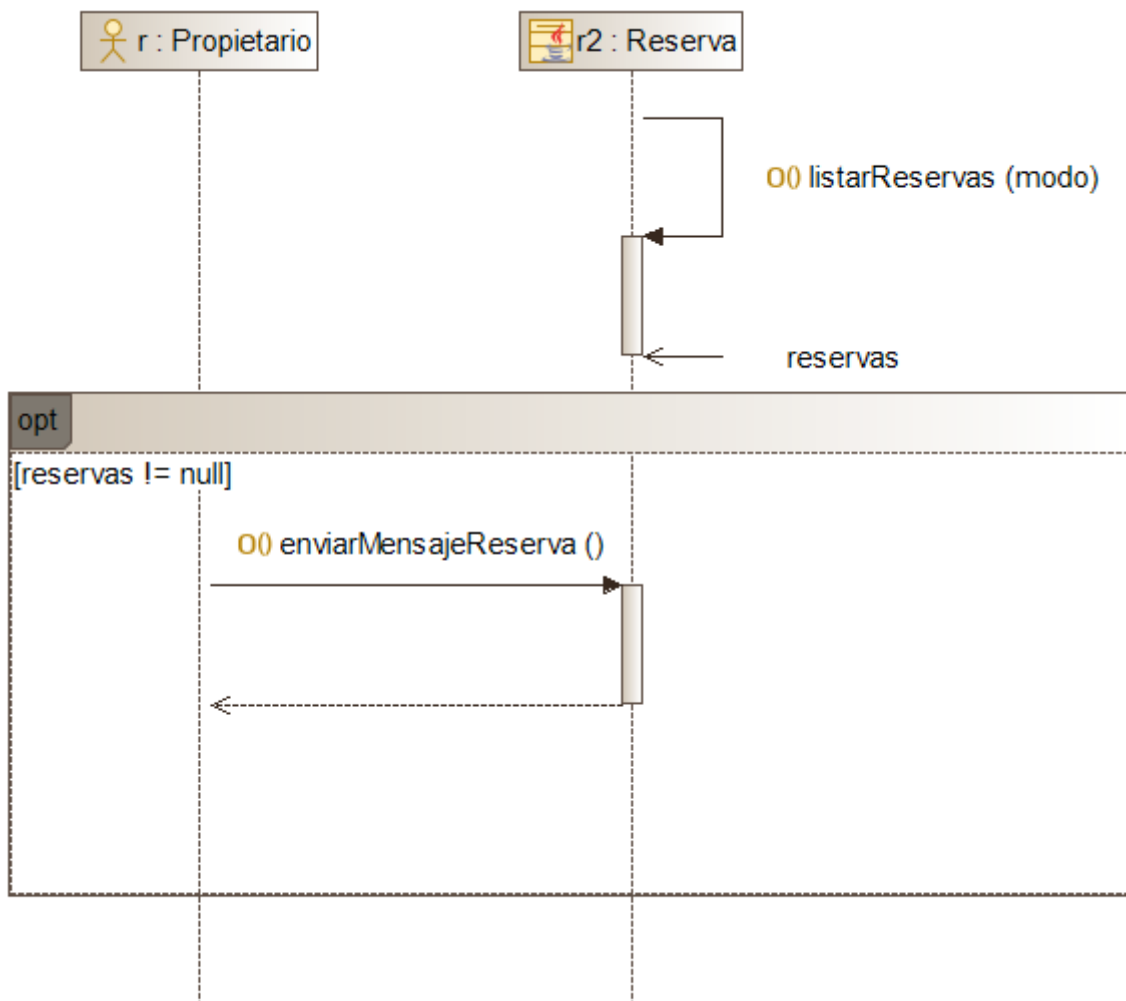


Fig. 12: Diagrama de secuencia del caso de uso "Confirmar reserva"

4. Prototipado del usuario.

4.1. Prototipos de pantallas.

Para el diseño del primer prototipo a papel, el cual se ha diseñado para un único dispositivo, el cual es dispositivos móviles (Android), siempre se va a mostrar en vertical.

Se ha intentado mantener la misma forma en todos los botones, y en aquellos campos en los que el usuario tenga que rellenar.

Para aquellas pantallas en las que haya que escribir algo, cuando se pulse la casilla correspondiente, saldrá un teclado, el cual ocupará aproximadamente la mitad inferior de la pantalla.

Las pantallas se muestran a continuación, sin seguir un orden concreto, ya que el orden de las pantallas se mostrarán en el mapa de navegación.



TUS RESERVAS

Orlando Muñiz

Eric Robles

Maria-

Modificar

Eliminar

Paco Pino

Amaia Lara

Antoni Arranz

Lara Falcon

Consulta de reserva

Enviar mensaje

Fecha de reserva

Nombre:

Juan García

Teléfono:

642 34 54 12

Datos del cliente

Entrada:

10/12/2022

Salida:

14/12/2022

Habitaciones reservadas

Habitación 1	1 persona
Habitación 2	3 personas
Habitación 3	5 personas
Habitación 4	2 personas
Habitación 5	1 persona
Habitación 6	9 personas

Total:

253,44€

TUS RESERVAS

Modifica la reserva

Datos del cliente

Nombre del cliente

Juan García

Teléfono

642 34 54 12

Fecha de reserva

Entrada

10/12/2022

Salida

14/12/2022

Cancelar

Continuar

Consulta de reserva

Fecha de reserva

Nombre:

Juan García

Teléfono:

642 34 54 12

Datos del cliente

Entrada:

10/12/2022

Salida:

14/12/2022

Habitaciones reservadas

Habitación 1	1 persona
Habitación 2	3 personas
Habitación 3	5 personas
Habitación 4	2 personas
Habitación 5	1 persona
Habitación 6	9 personas

Total:

253,44€

Añade una reserva

Datos del cliente

Nombre del cliente

Añade aquí el nombre del cliente

Teléfono

Ej. 662 98 07 18

Fecha de reserva

Entrada

Salida

Cancelar

Continuar

Habitaciones reservadas

Precio total: 127,99€

Habitación 1.1

Habitación 1.2

Habitación 1.3

Habitación 2.1

Habitación 2.2

Habitación 2.3

Habitación 3.1

Habitaciones reservadas

Precio total: 0,00 €

Añadir a reserva

Habitación 1.1

Habitación 1.2

Habitación 1.3

Habitación 2.1

Habitación 2.2

Habitación 2.3

Habitación 3.1

Añadir a reserva

Habitación 1.1

Habitación 1.2

Ocupantes (máx. 20)

Cancelar Añadir

Habitación 2.2

Habitación 2.3

Habitación 3.1

TUS HABITACIONES

Habitación 1.1

Habitación 1.2

Habitación 1.3

Habitación 2.1

Habitación 2.2

Habitación 2.3

Habitación 3.1

TUS

Añadir habitación

Ordenar por identificador

Ordenar por numero de ocupantes

Ordenar por precio

Habitación 1.1

Habitación 1.2

Habitación 1.3

Habitación 2.1

Habitación 2.2

Habitación 2.3

Habitación 3.1

⊕

Modificar habitación

⊖

TUS HABITACIONES

⋮

Identificador de la habitación

Habitación 1.1

Descripción

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean euismod bibendum laoreet. Proin gravida dolor sit amet lacus accumsan et viverra justo commodo. Proin sodales pulvinar sic tempor. Sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nam fermentum, nulla luctus pharetra vulputate, felis tellus mollis orci, sed rhoncus pronin sapien nunc accuan eget.

Número de ocupantes

13

Precio por noche para un solo ocupante

Recargo por cada ocupante extra

15

Cancelar

Modificar

Habitación 1.1

Habitación 1.2

Habitación 2.1

Habitación 2.2

Habitación 2.3

Habitación 3.1

Descripción

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean euismod bibendum laoreet. Proin gravida dolor sit amet lacus accumsan et viverra justo commodo. Proin sodales pulvinar sic tempor. Sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nam fermentum, nulla luctus pharetra vulputate, felis tellus mollis orci, sed rhoncus pronin sapien nunc accuan eget.

Número de ocupantes

13

Precio por noche para un solo ocupante

15,99 €

Recargo por cada ocupante extra

15%

⊕

TUS HABITACIONES

⋮

⊕

Añade una habitación

Identificador de la habitación

Añade aquí el nombre de la habitación

Descripción

Describe tu habitación

Número de ocupantes

Precio por noche para un solo ocupante

Recargo por cada ocupante extra

% del precio

Cancelar

Añadir

4.2. Mapa de navegación del sistema

A continuación, se va a realizar el mapa de navegación de nuestro sistema y se va a explicar cómo funciona.

Se ha puesto un círculo rojo al principio del mapa que indica el comienzo del mapa de navegación.

Por ejemplo, si gestionamos reservas, y no hay ninguna reserva, nos aparecerá la pantalla P2 sin ninguna reserva hecha (sería una pantalla vacía en la que solo podrías añadir reservas). (La primera vez que se entra en la aplicación). Lo mismo pasaría con las habitaciones.

En la pantalla P2, cuando se pulsa los tres puntos, se desplegará un recuadro arriba a la derecha (P3). También sucede con la pantalla P5 y P6, y con la pantalla P13 y P15, respectivamente.

En la pantalla P2, si se presiona sobre un nombre de la reserva durante dos segundos, se mostrará un pequeño recuadro justo abajo del nombre presionado (P4). Si pulsamos normal, nos redirigirá a P5. En la pantalla P14 sucederá lo mismo que en P2, si se presiona sobre un nomb

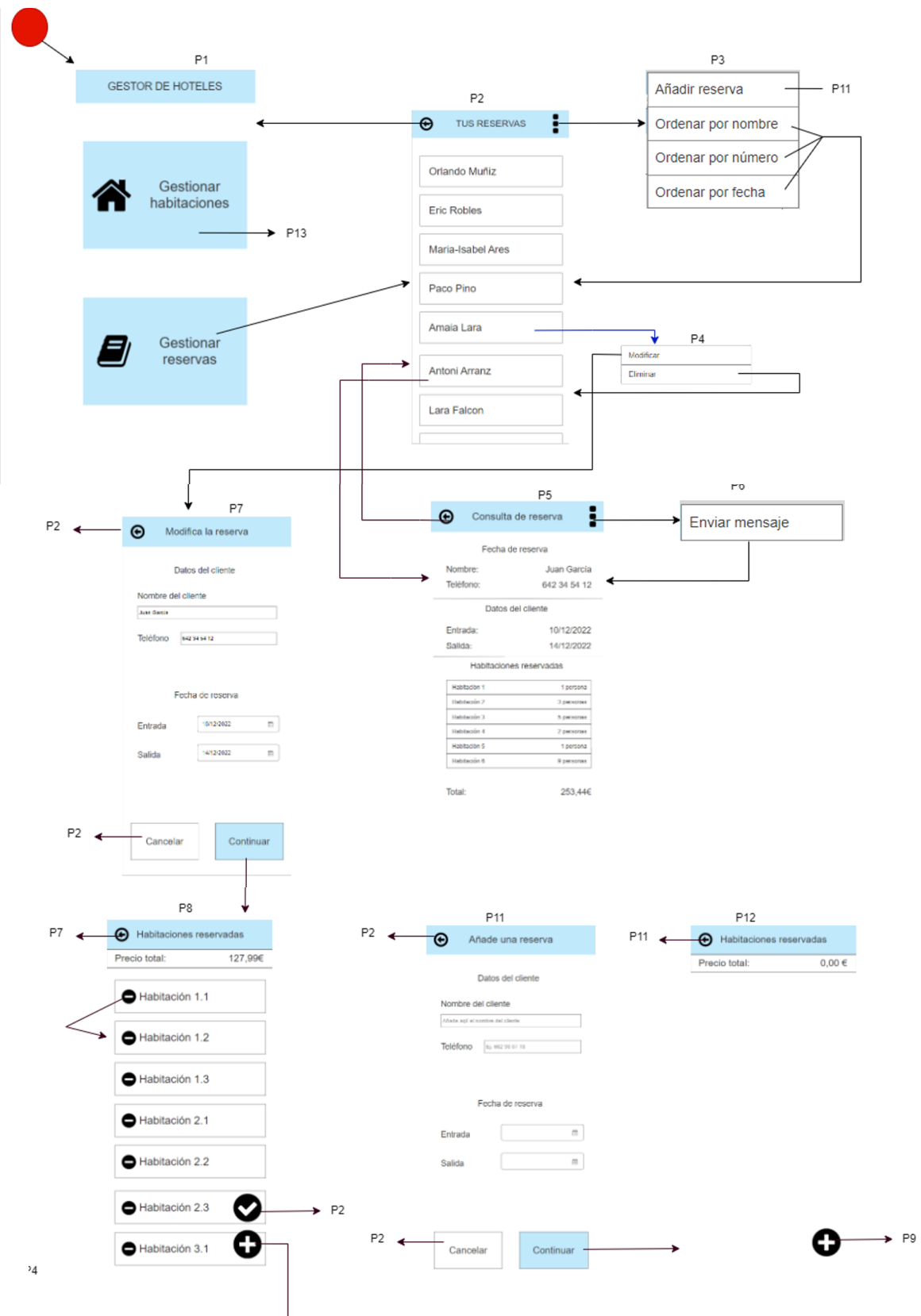
Si se elimina una reserva de la lista de reservas, te redirigirá otra vez a la lista con esa reserva eliminada.

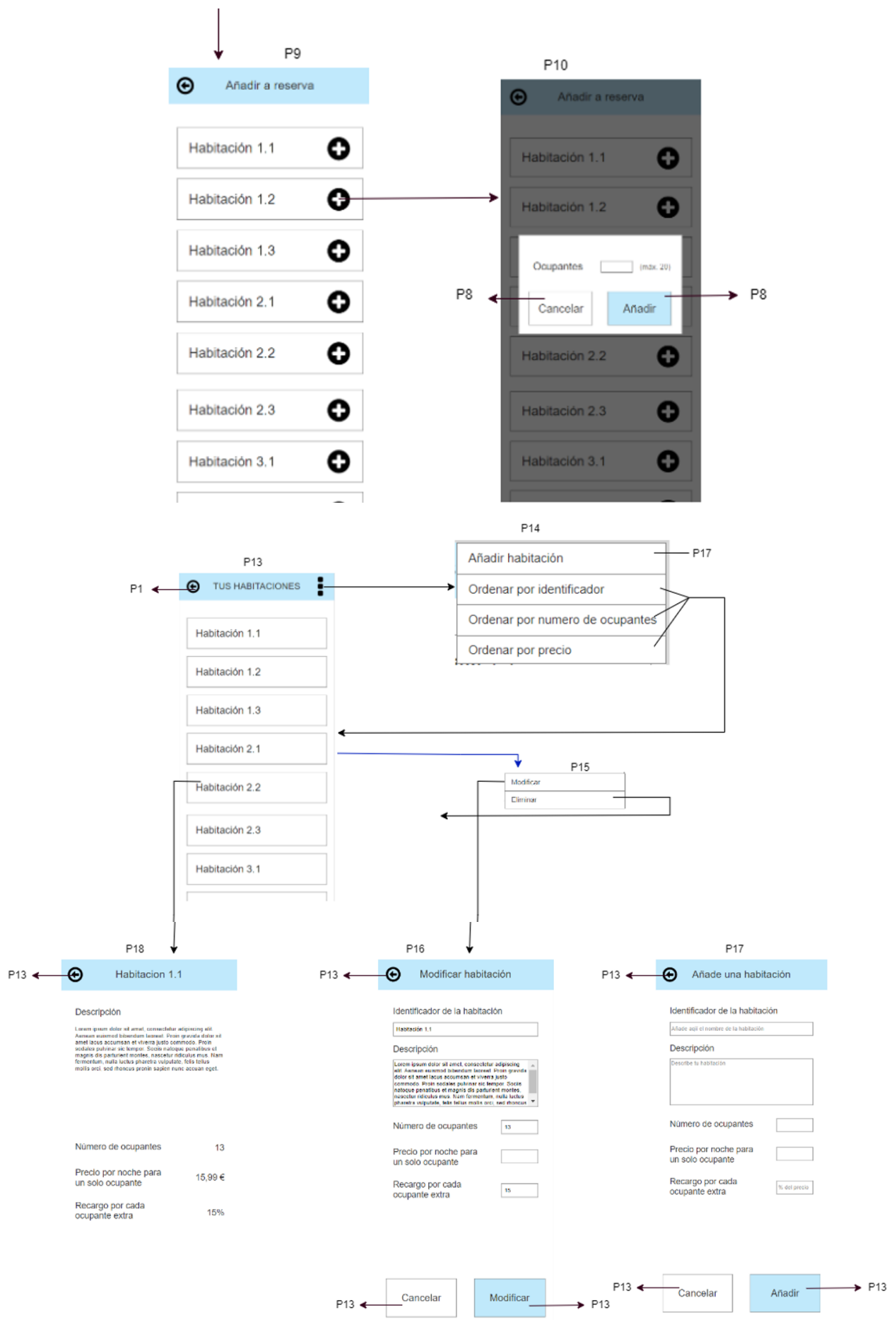
En P8, cuando se elimina una habitación (presionar el icono del menos) de la reserva, se eliminará de la pantalla la habitación reservada.

Si en la pantalla P10 y P11 le damos a confirmar, se confirmará la operación de realizar reserva y volverán a P8 con su reserva correspondiente. Si le damos a cancelar, no habrá hecho nada y volverá a la pantalla P8.

Cada vez que le demos a un recuadro (un campo donde escribir), se desplegará el teclado de android justo en la parte de abajo de la pantalla. Cuando se elija las fechas de salida y de llegada, se desplegará un calendario en el que se podrá elegir día, mes y año.

Y si en algún momento, se confirma con algún campo de texto sin rellenar, nos saldrá un error de "Faltan campos".





5. Diseño del sistema y diseño de objetos

5.1. Modelo lógico de la base de datos

Se va a realizar el modelado lógico de la base de datos relacional. Primero vamos a realizar el esquema E/R en el que hay dos tablas, Habitación y Reserva y una relación N:1 para relacionarlos.

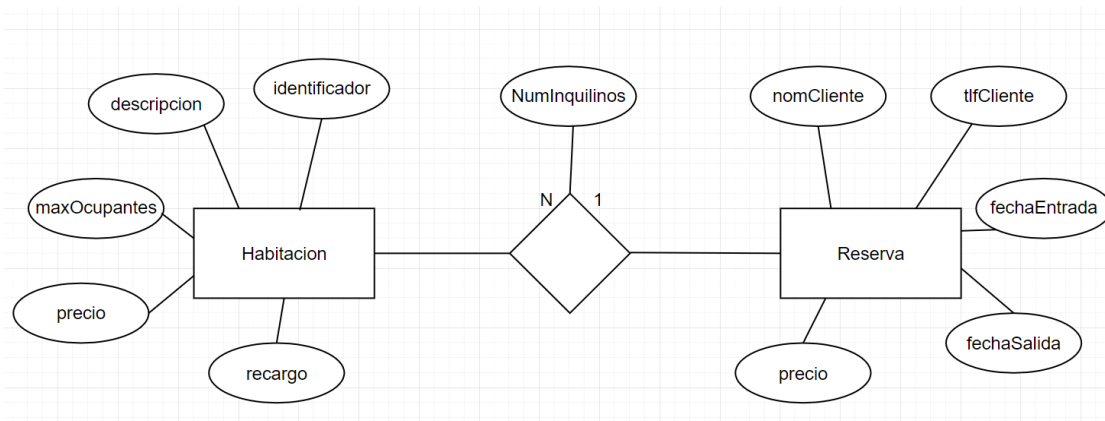


Fig. 14: Diagrama entidad-relación de la aplicación

Hemos decidido hacer la relación “HabitacionesReservadas”, una entidad que tenga la tupla de claves primarias de Habitación y Reserva respectivamente.

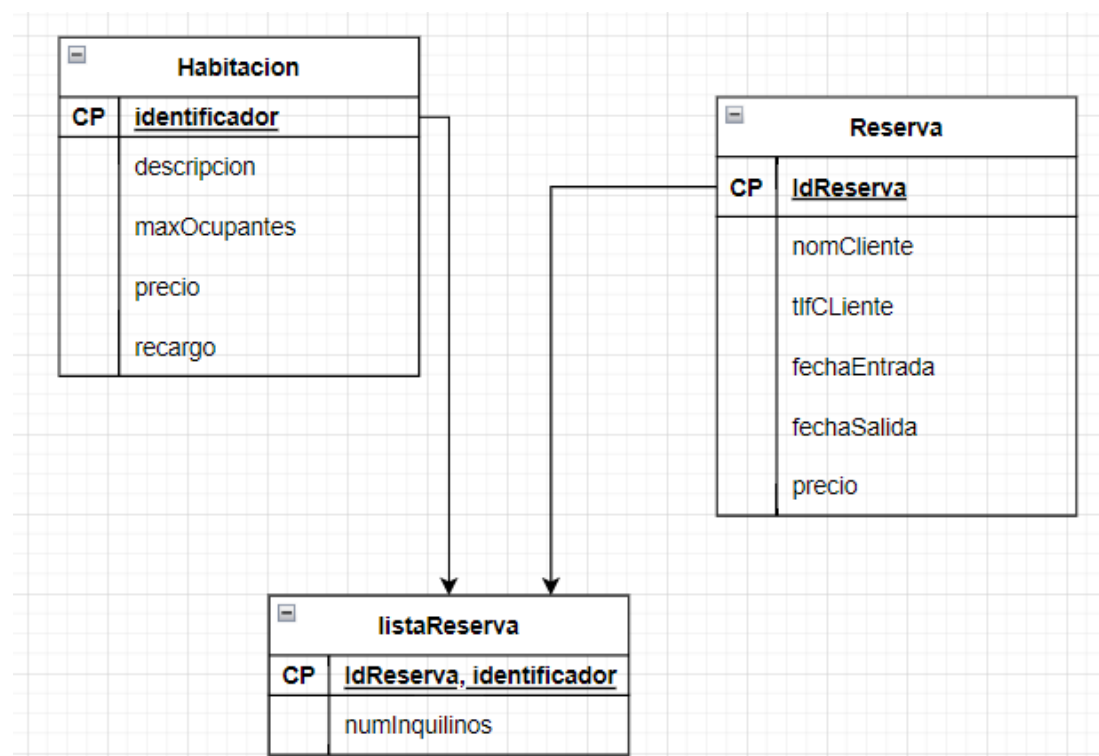


Fig. 15: Diagrama del modelo relacional
reserva_id, identificador_id

Se van a escribir las sentencias SQL de la base de datos:

```
CREATE TABLE Habitacion (  
    _ID                integer not null primary key autoincrement,  
    descripcion        int not null,  
    maxOcupantes       int not null,  
    precio             real not null,  
    recargo            real not null,  
);
```

```
CREATE TABLE Reserva (  
    _ID                int primary key autoincrement  
    nomCliente         VARCHAR(50) PRIMARY KEY,  
    tlfCliente         int not null,  
    fechaEntrada       text not null,  
    fechaSalida        text not null,  
    precio             real not null  
);
```

```
CREATE TABLE ListaReserva(  
    numInquilinos      int not null  
    identificador      text not null references Habitacion(identificador),  
    IdReserva          int not null references Reserva(IdReserva),  
    primary key(identificador, IdReserva)  
);
```

5.2. Diagrama de clases a nivel de diseño

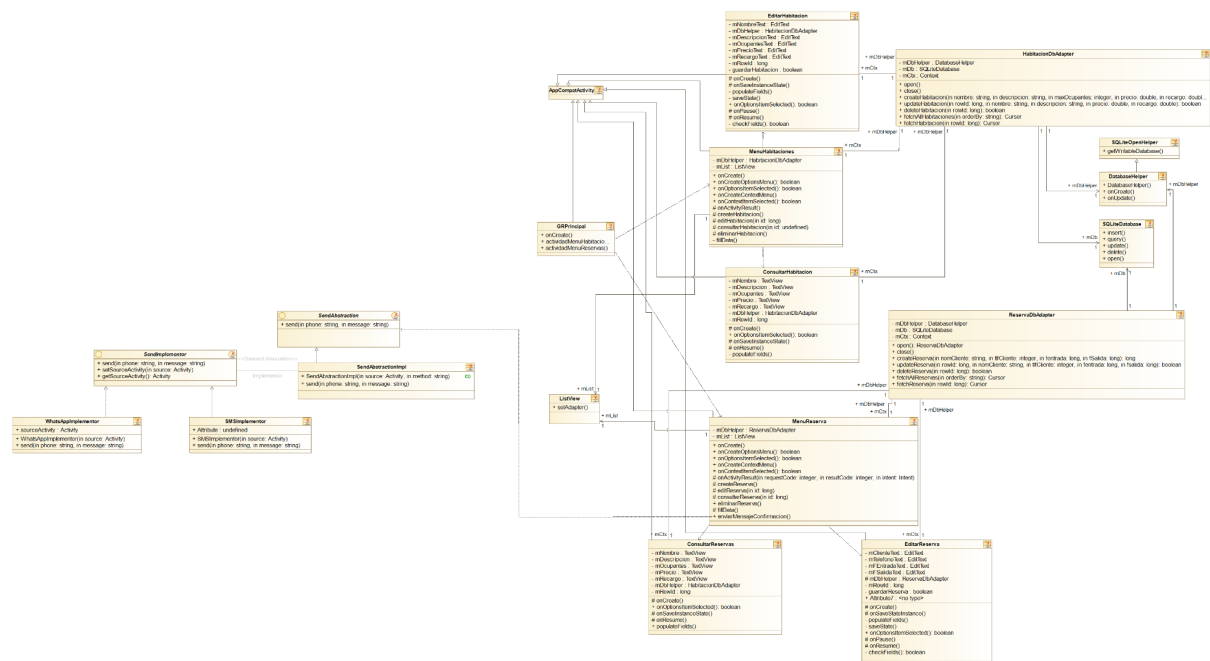


Fig. 16: Diagrama de clases a nivel de diseño de la aplicación

Con ayuda del fichero notepad3, se ha realizado este diagrama de clases. Se ha creado un adaptador para las habitaciones y otro para las reservas. Hemos conectado las respectivas clases con la base de datos sqlite. Este diagrama de clases se ha utilizado como índice para realizar los demás diagramas.

5.3. Diagrama de paquetes

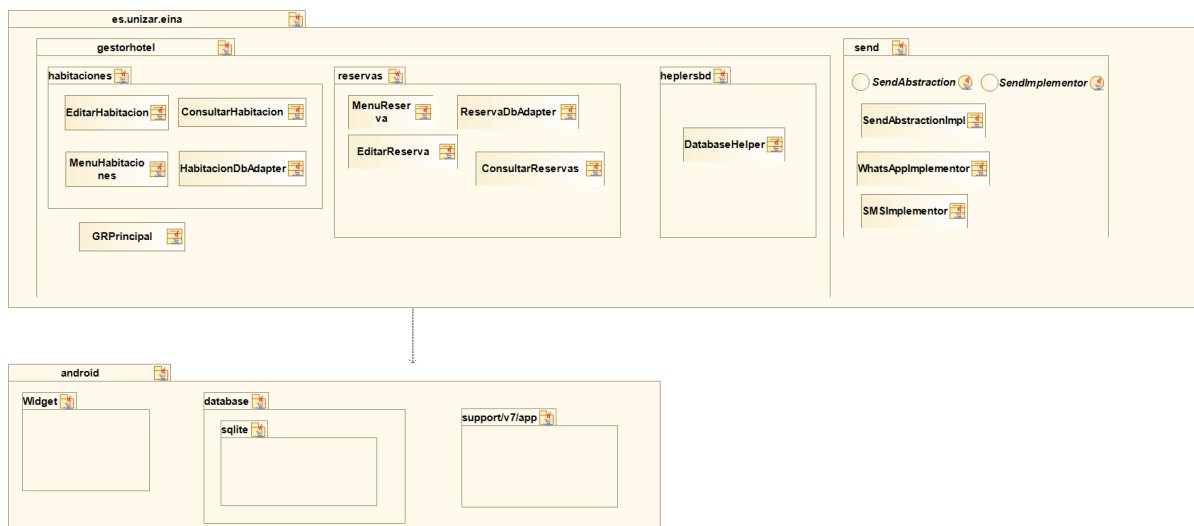


Fig. 17: Diagrama de paquetes de la aplicación

Respecto al diagrama de clases, hemos realizado los paquetes. Por un lado las interfaces de reservas, habitaciones y lo respectivo a ello, y por otra parte, la base de datos. gestorhotel y send como dos diferentes,

5.4. Diagrama de componentes



Fig. 18: Diagrama de componentes de la aplicación

La aplicación se compone del gestor de hoteles y las funciones del emulador ,y luego la base de datos.

5.5. Diagrama de despliegue

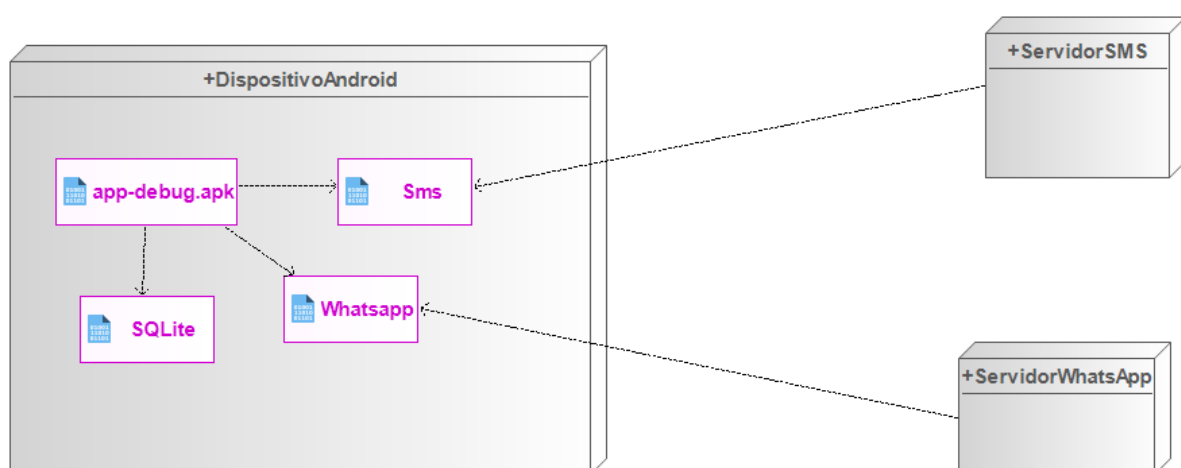


Fig. 19: Diagrama de despliegue de la aplicación

El diagrama de despliegue del dispositivo android es que la aplicación está en un fichero apk que se ejecutaría y por otra parte el SQLite. **aquí ita el com.sms y el com.whatsapp desde app-debug.apk (como sqlite) dos nodos igual que dispositivoAndroid que sean servidor de whatsapp y sms**

5.6. Diagramas de secuencia a nivel de diseño

Caso de uso: Añadir habitación

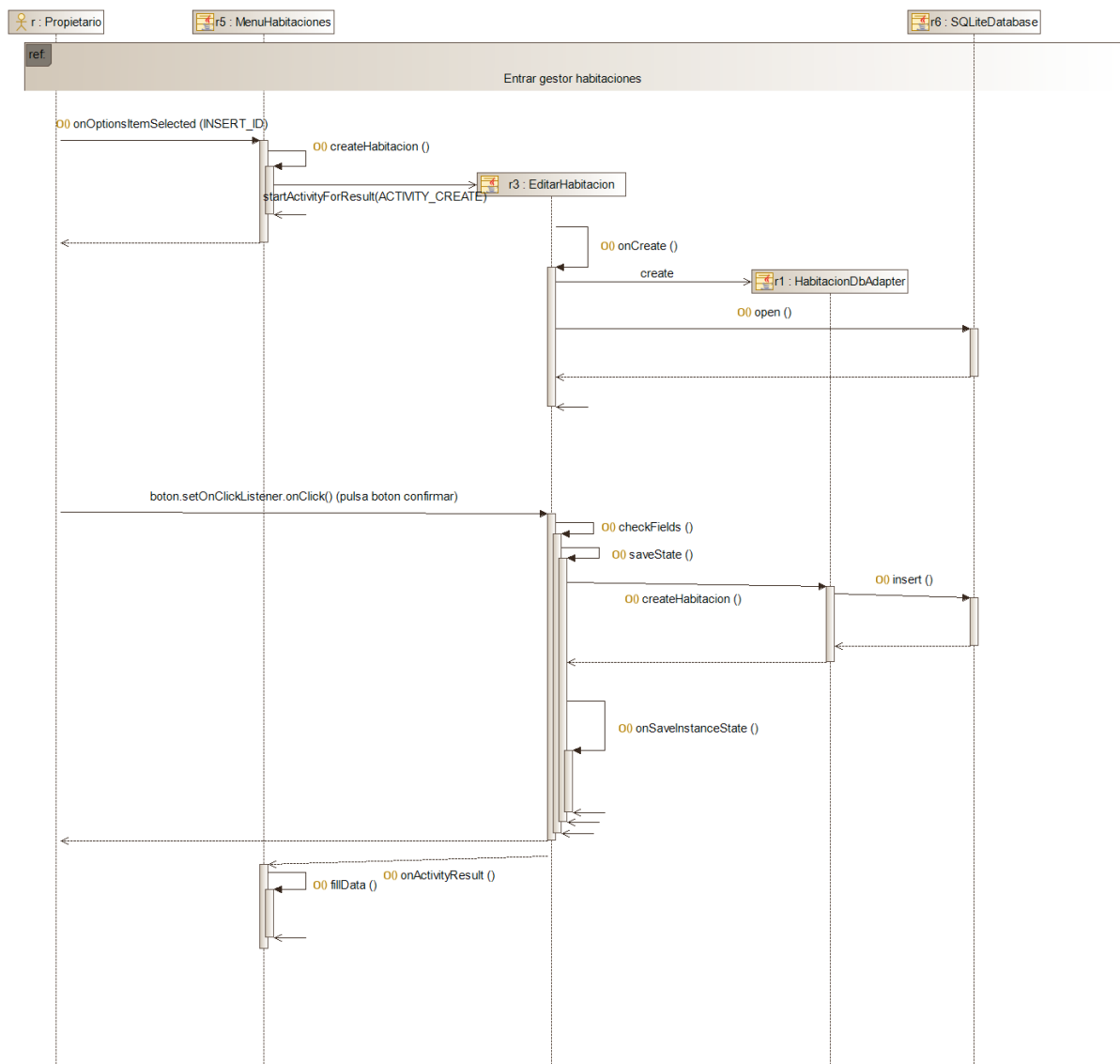


Fig. 20: Diagrama de secuencia para añadir una habitación a nivel de diseño

Caso de uso: Eliminar habitación

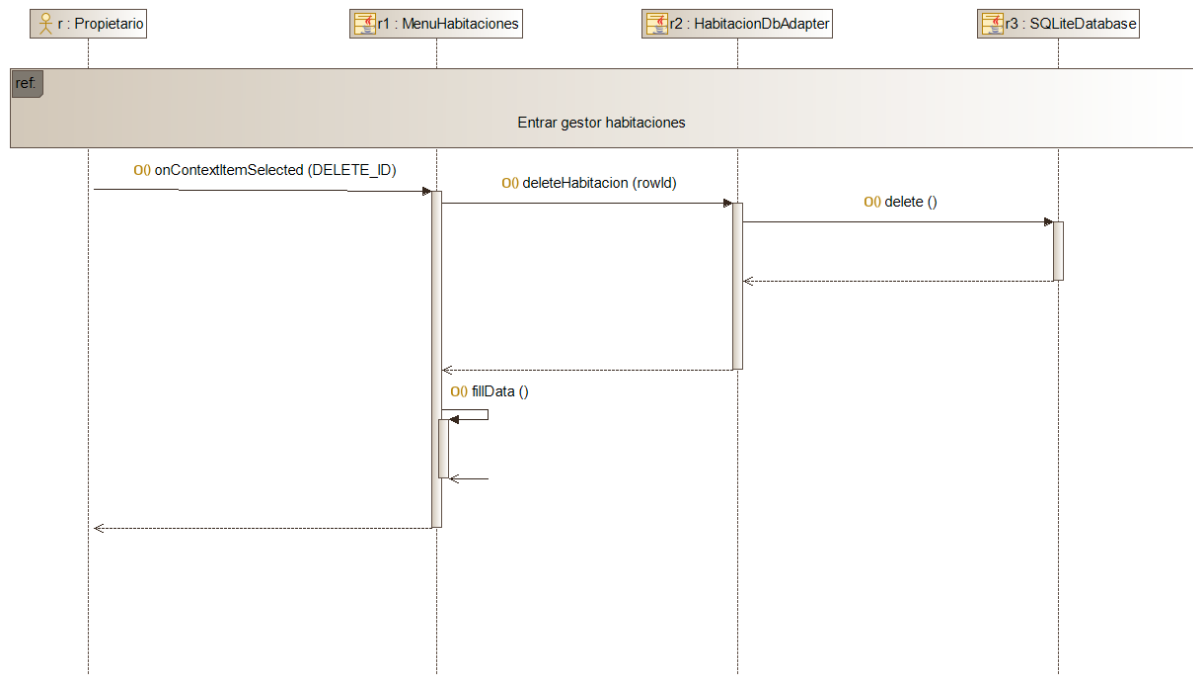


Fig. 21: Diagrama de secuencia para eliminar una habitación a nivel de diseño

Caso de uso: Modificar habitación

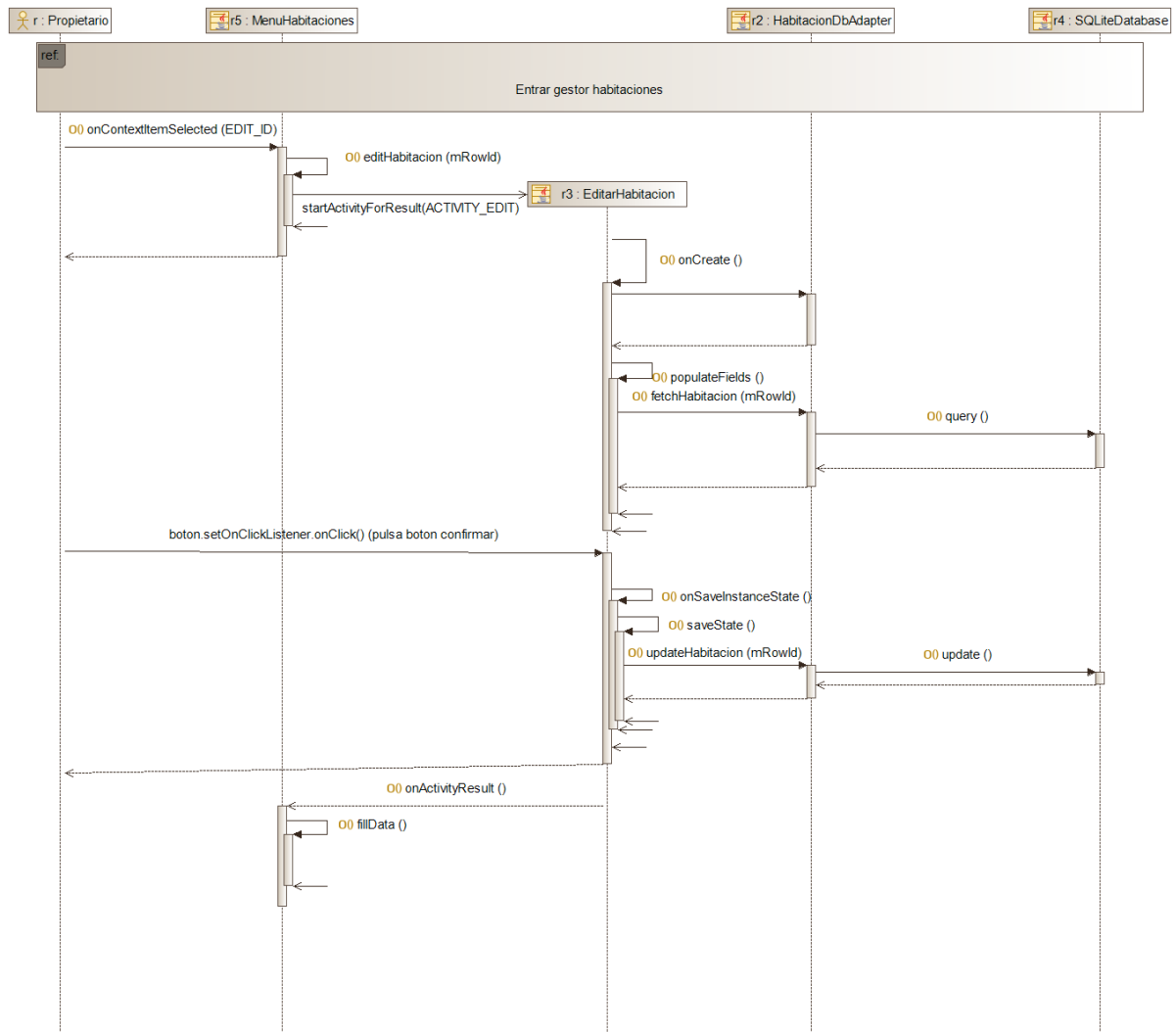


Fig. 22: Diagrama de secuencia para modificar una habitación a nivel de diseño

Caso de uso: Consultar habitación

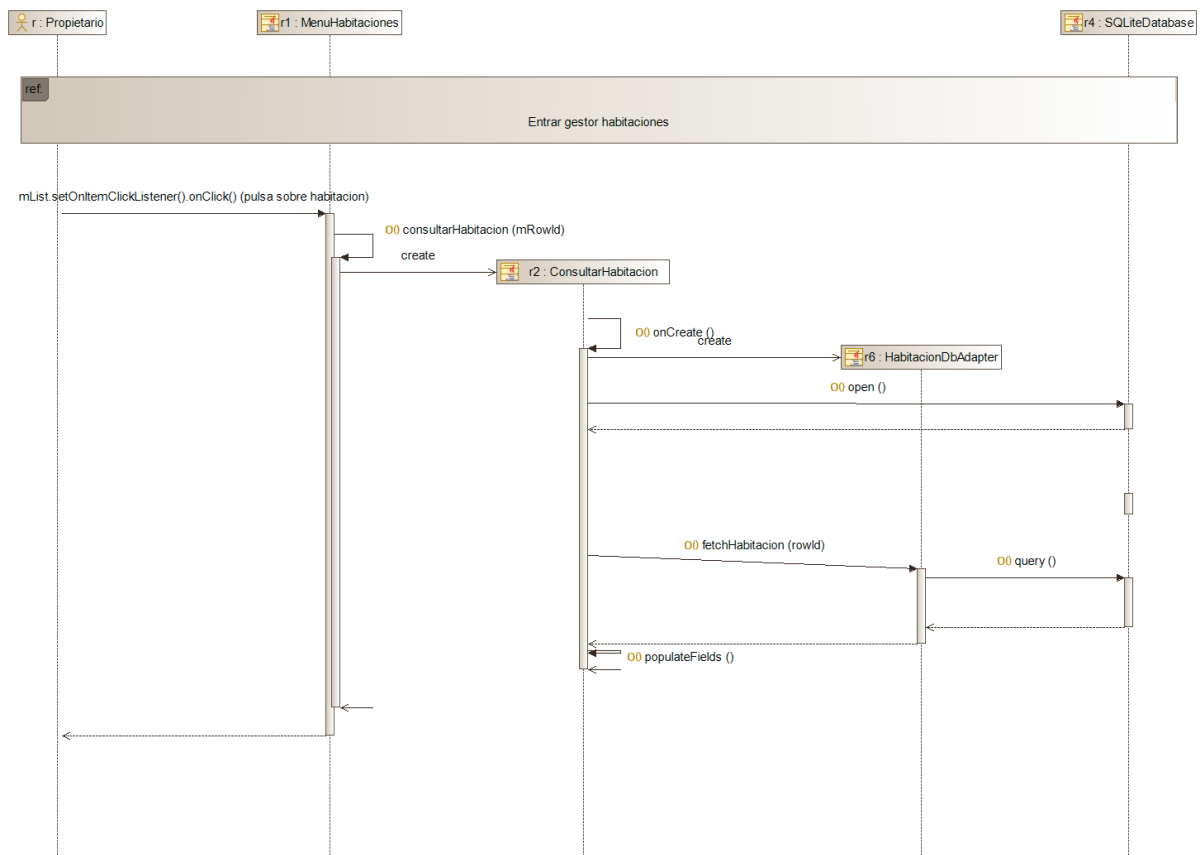


Fig. 23: Diagrama de secuencia para consultar una habitación a nivel de diseño

Caso de uso: Añadir reserva

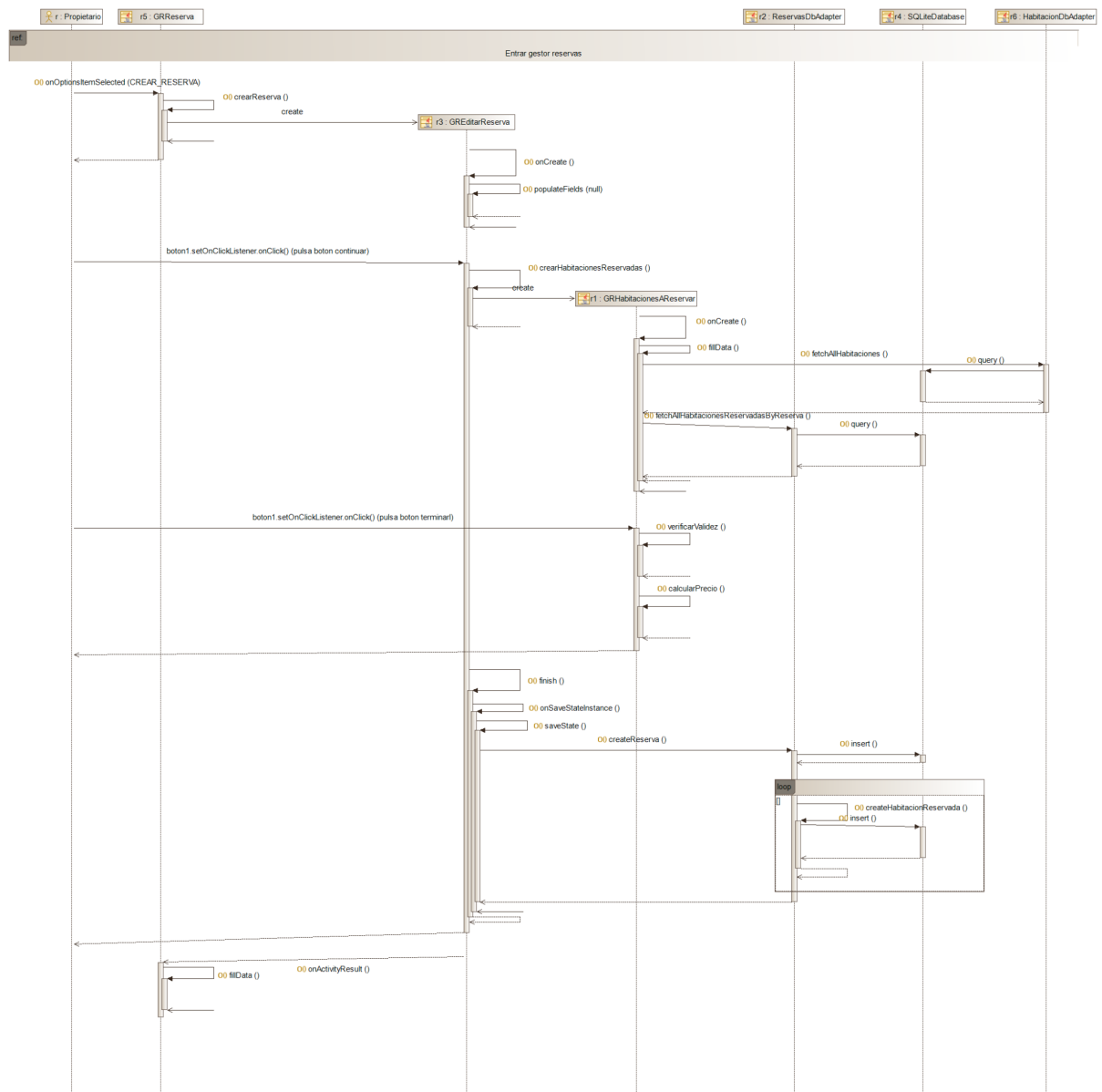


Fig. 24: Diagrama de secuencia para añadir una reserva a nivel de diseño

Caso de uso: Eliminar reserva

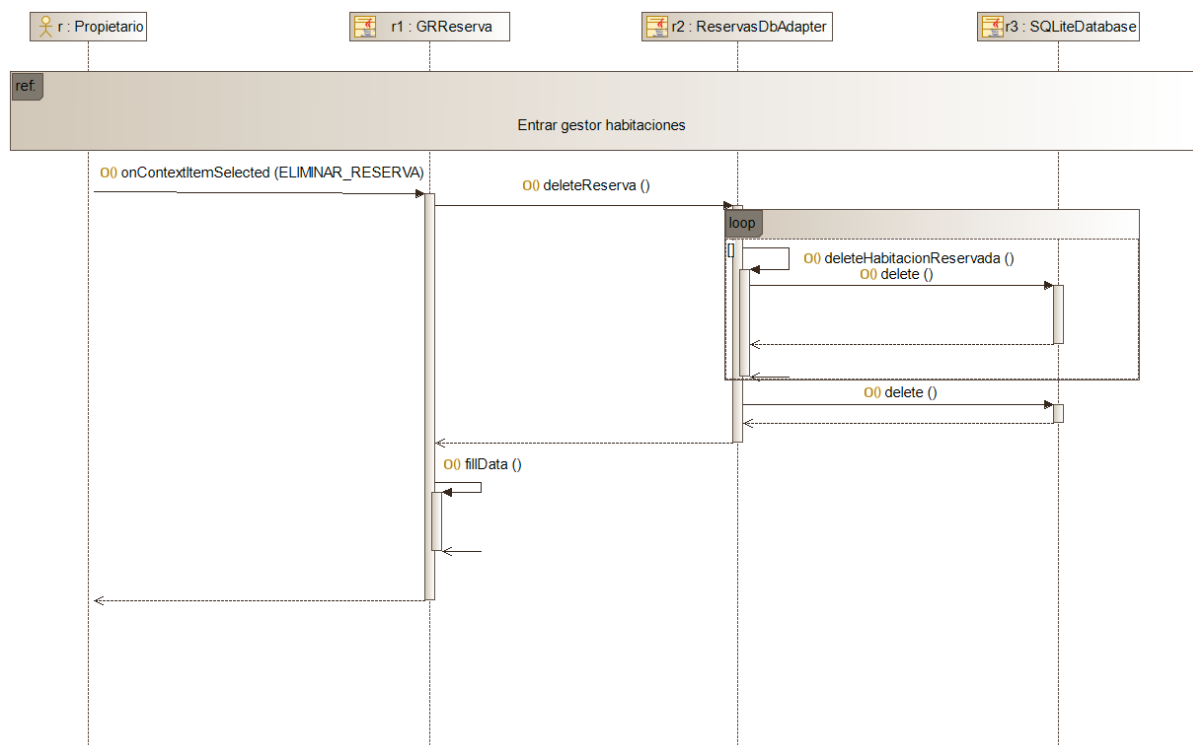


Fig. 25: Diagrama de secuencia para eliminar una reserva a nivel de diseño

Caso de uso: Modificar reserva

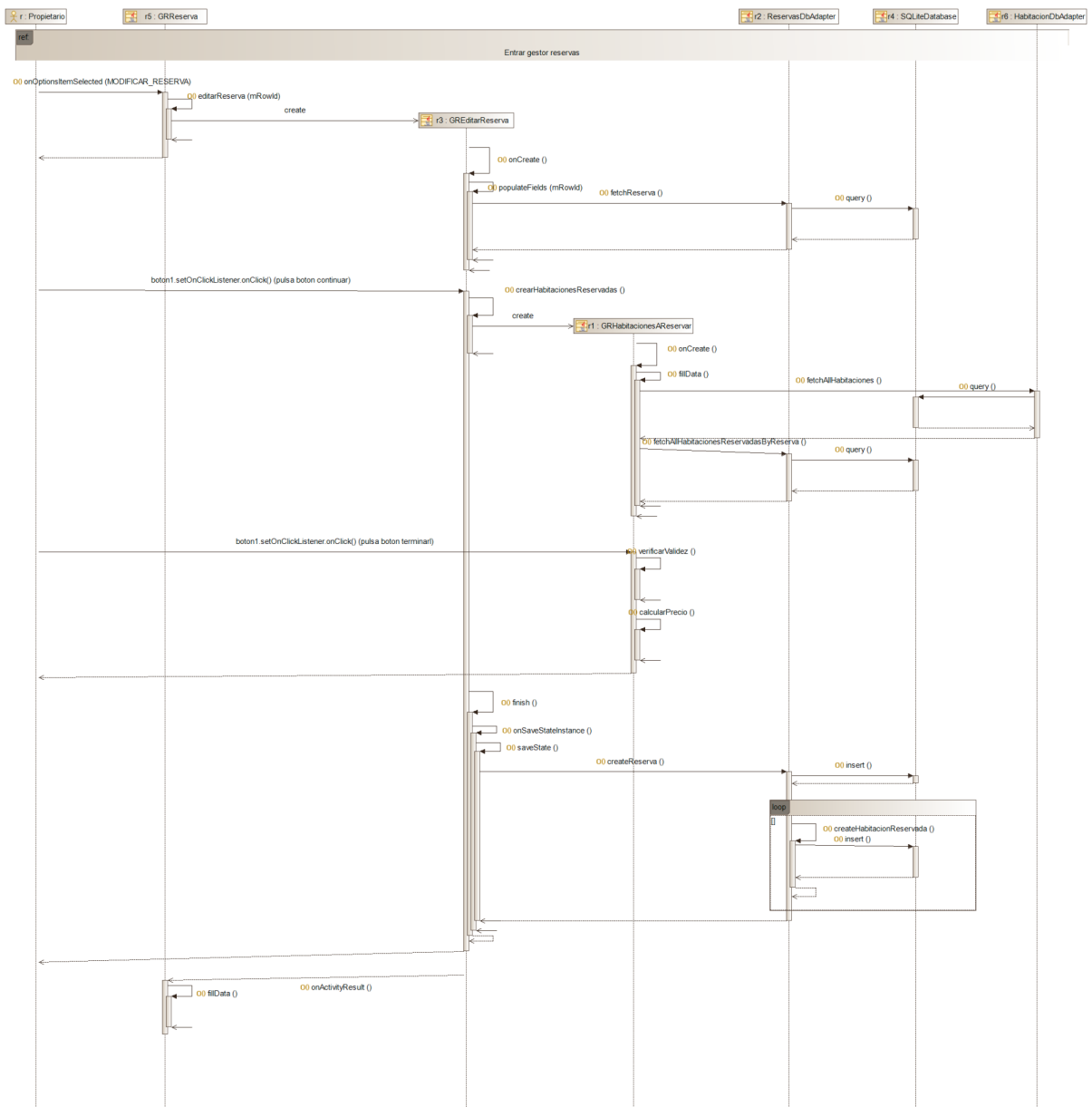


Fig. 26: Diagrama de secuencia para modificar una reserva a nivel de diseño

Caso de uso: Consultar reserva

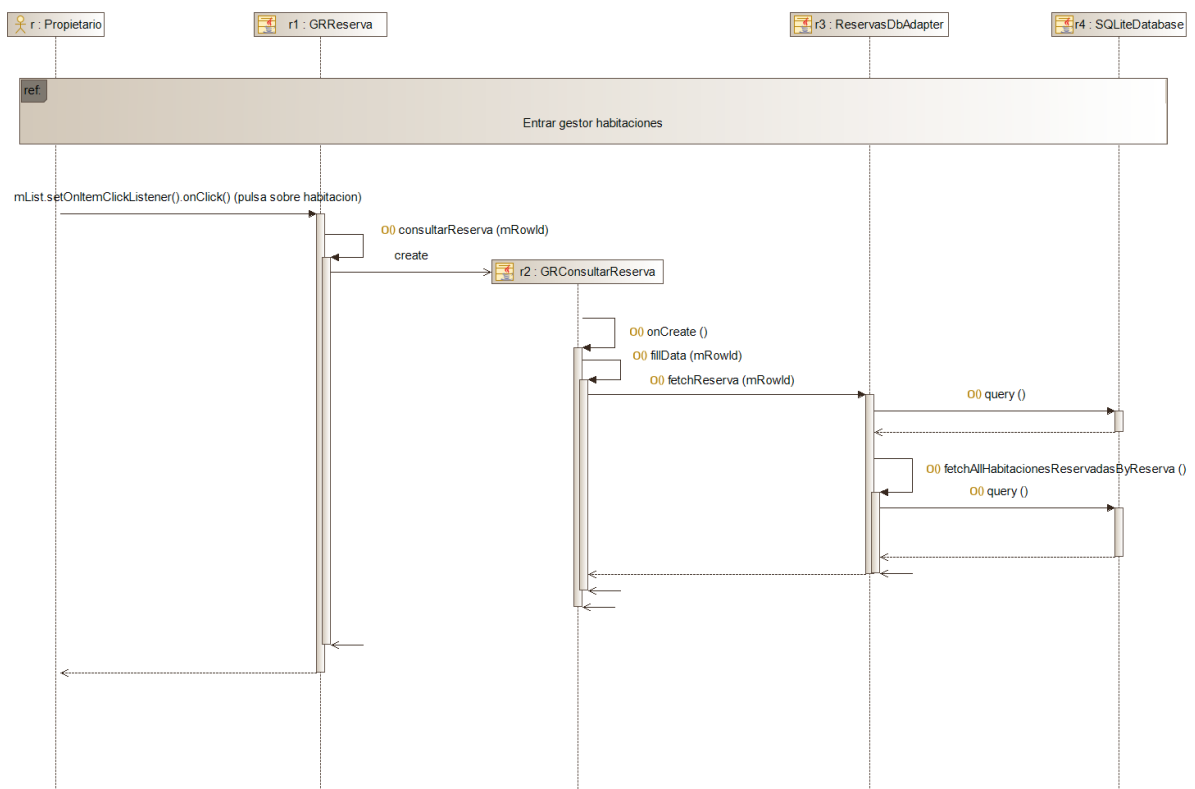


Fig. 27: Diagrama de secuencia para consultar una reserva a nivel de diseño

Caso de uso: Enviar mensaje de confirmación

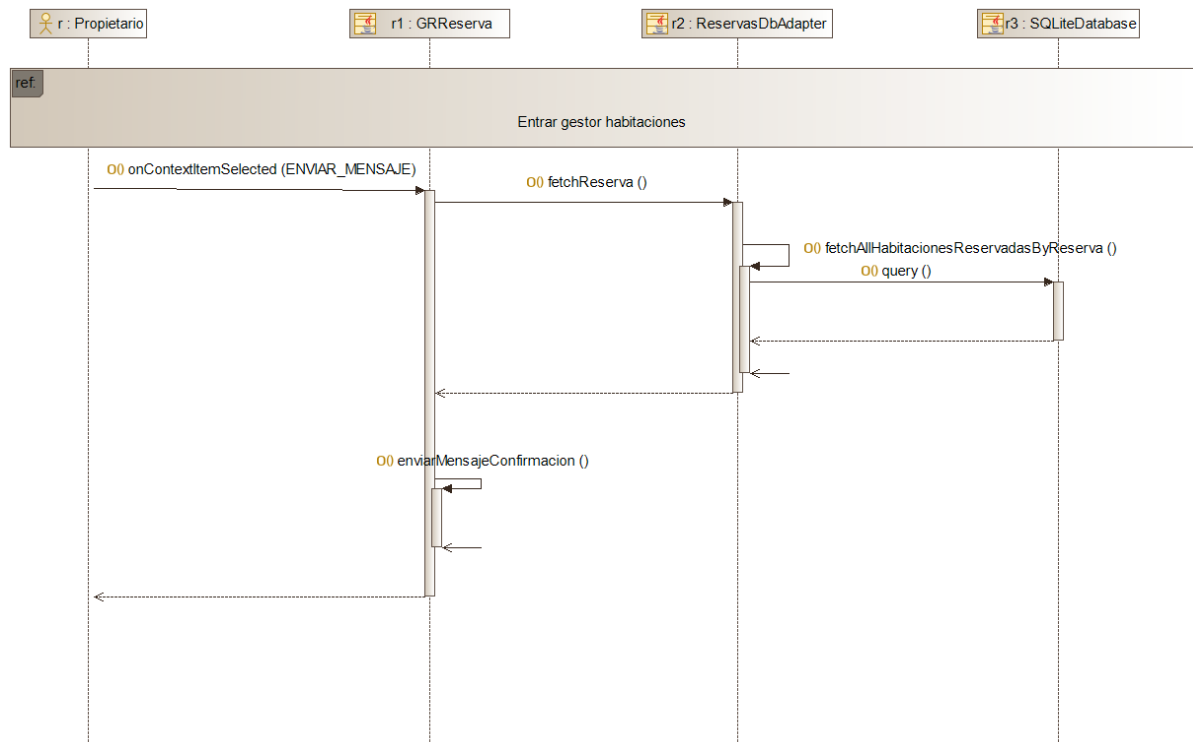


Fig. 28: Diagrama de secuencia para enviar un mensaje de confirmación de una reserva a nivel de diseño

```
sequenceDiagram
    participant r1 as r1 : Proprietario
    participant r2 as r2 : GRReserva
    participant r3 as r3 : ReservasDbAdapter
    participant r4 as r4 : SQLiteDatabase

    r1->>r2: botonhab.setOnClickListener.onClick() (puls a opción gestor reservas)
    activate r2
    r2->>r2: 00 gestorArReservas ()
    r2->>r2: create
    r2->>r2: 00 onCreate ()
    r2->>r3: create
    activate r3
    r3->>r3: 00 open ()
    r3->>r2: 00 fillData ()
    deactivate r3
    r2->>r2: 00 fetchAllReservas ()
    r2->>r4: 00 query ()
    activate r4
    r4-->>r2: 
    deactivate r4
    r2-->>r1: 
    deactivate r2

    opt opt
        r1->>r2: 00 onItemSelected (mode)
        activate r2
        r2->>r2: 00 fillData ()
        activate r2
        alt all
            [mode = BY_NOMBRE]
            r2->>r3: 00 fetchAllReservas ()
            activate r3
            r3->>r4: 00 query ()
            activate r4
            r4-->>r3: 
            deactivate r4
            r3-->>r2: 
            deactivate r3
        and [mode = BY_TELEFONO]
            r2->>r3: 00 fetchReservasByTelefono ()
            activate r3
            r3->>r4: 00 query ()
            activate r4
            r4-->>r3: 
            deactivate r4
            r3-->>r2: 
            deactivate r3
        and [mode = BY_FECHA]
            r2->>r3: 00 fetchAllReservasByFecha ()
            activate r3
            r3->>r4: 00 query ()
            activate r4
            r4-->>r3: 
            deactivate r4
            r3-->>r2: 
            deactivate r3
        end
        r2-->>r1: 
        deactivate r2
    end
```

44

Subdiagrama: Entrar en gestor de habitaciones

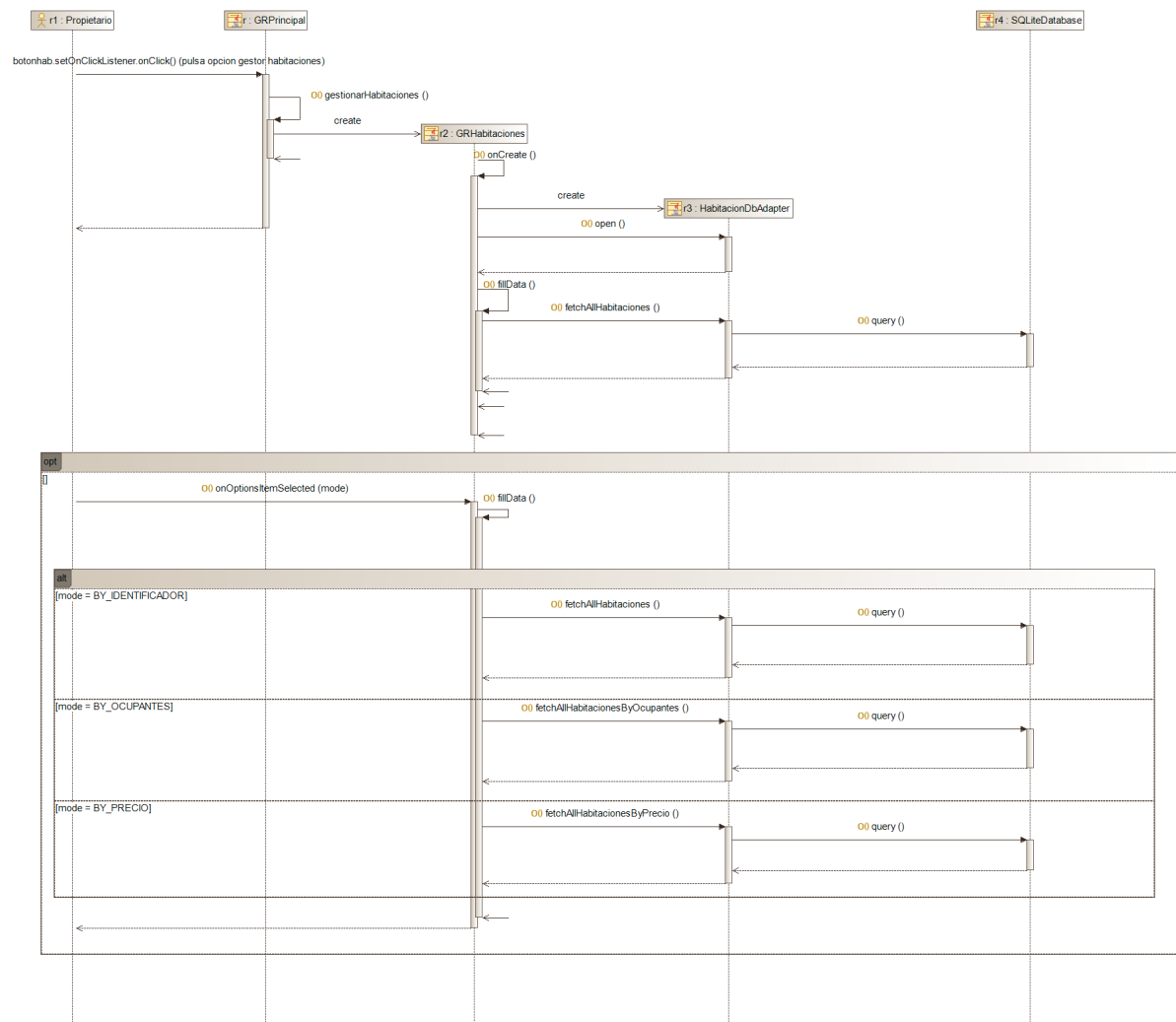


Fig. 30: Diagrama de secuencia auxiliar para abrir el menú de habitaciones

Anexo: Horas dedicadas

Tiempo (h)	Dorian	Sergio	Total
Requisitos	1.5	1.5	3
Modelado estático	4	4	8
Modelado dinámico	6	5	11
Mapa de navegación	2	6.5	8.5
Prototipo de pantallas	5	4.5	9.5
Memoria	7.5	11	18.5
Modelo BD	1.5	4	5.5
Diagramas a nivel de diseño	9	9	18
Implementación	12	4.5	16.5
Total	48.5	50	99.5