Cairo University  
Faculty of Computers and Artificial Intelligent

**CS251**

**Software** **Engineering I**

Hotel System

Software Design

August 2022

Contents

[Instructions [To be removed] 3](#_Toc101814919)

[Team 3](#_Toc101814920)

[Document Purpose and Audience 3](#_Toc101814921)

[System Models 3](#_Toc101814922)

[I. Class diagrams 3](#_Toc101814923)

[Important Algorithm 4](#_Toc101814924)

[II. Sequence diagrams 5](#_Toc101814925)

[Class - Sequence Usage Table 6](#_Toc101814926)

[Ownership Report 6](#_Toc101814927)

[Policy Regarding Plagiarism: 7](#_Toc101814928)

# Team

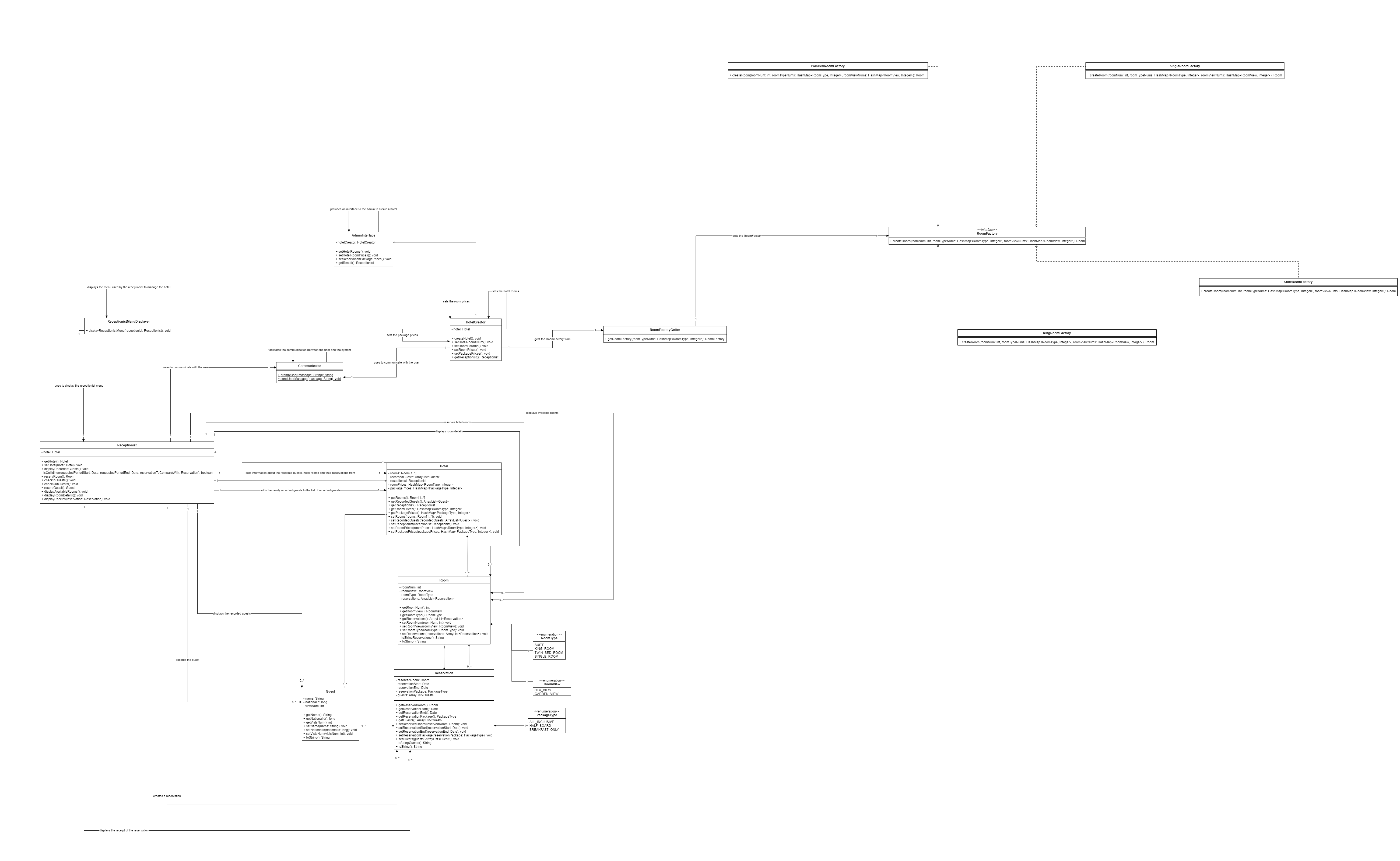
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20196006 | Ahmed mohamed Ahmed | 20196006@stud.fci-cu.edu.eg | 01272720920 |
| 20196115 | Omar Mohamed Elzohry | 20196115@stud.fci-cu.edu.eg | 01140665220 |

# Document Purpose and Audience

Purpose: the document describes the internal structure of the hotel system

Audience: developers, project managers and testers

# System Models

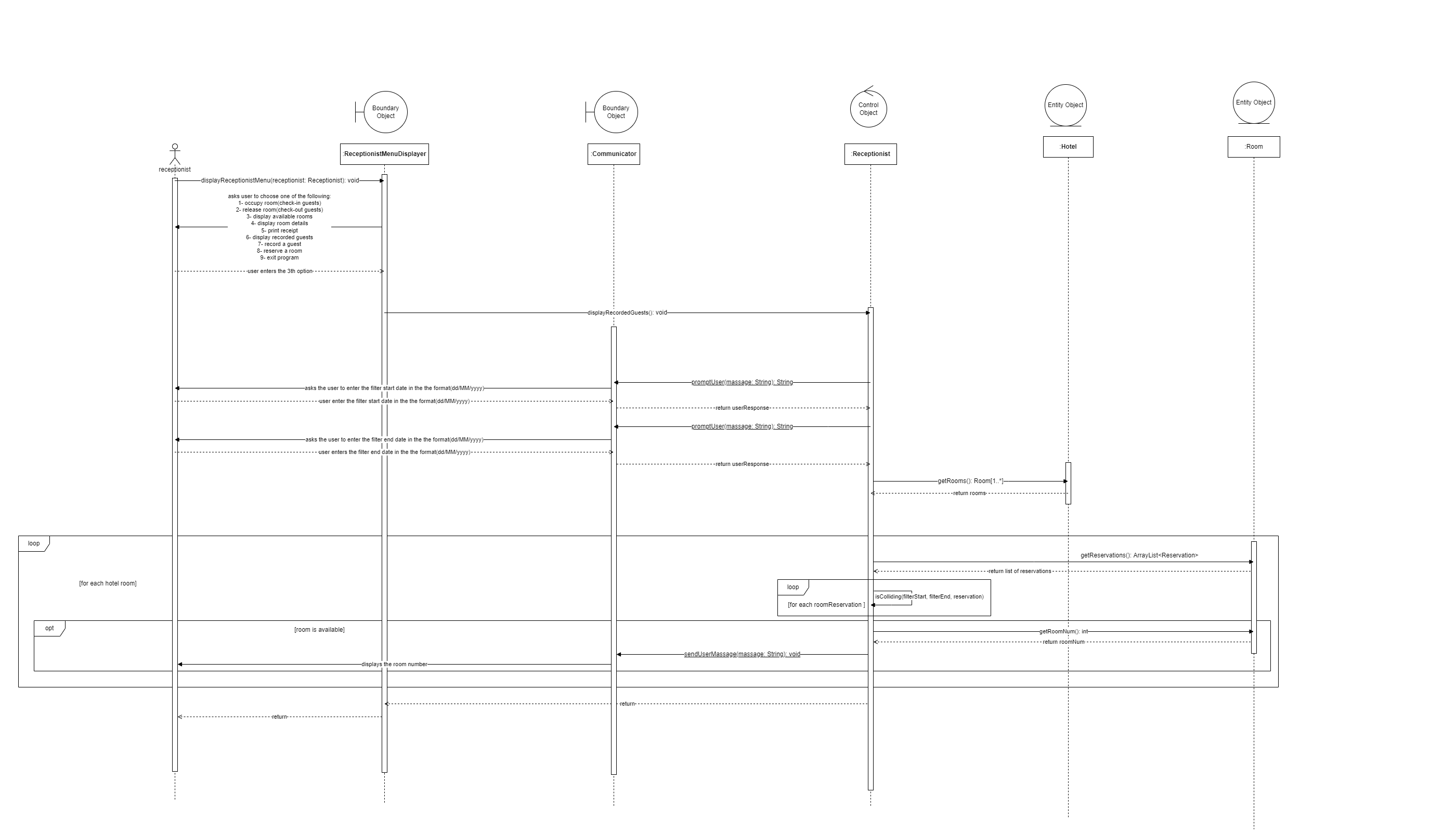


| **Class ID** | **Class Name** | **Description & Responsibility** |
| --- | --- | --- |
| 1 | ReceptionistMenuDisplayer | displays the menu that will be used by the Receptionist to manage the hotel |
| 2 | AdminInterface | Provides an interface used by the admin to initialize the hotel |
| 3 | Communicator | facilitates the communication between the user and the system |
| 4 | HotelCreator | Initializes the hotel parameters |
| 5 | Hotel | Stores information about the hotel such as the hotel rooms, prices of different types of rooms and prices of different reservation packages and is managed by the receptionist |
| 6 | Room | Stores the detailed information of a hotel room such as room number, room view, type of room and the reservation details of a room |
| 7 | Reservation | Stores information about hotel room reservations such as the reservation start and end dates and the type of package included with the reservation and the guests who booked the reservation |
| 8 | Guest | Stores information about the guests such as their names, national id and number of visits |
| 9 | Receptionist | Manages the hotel rooms and records information about guests |
| 10 | RoomFactoryGetter | Gets the requested room factory that will build the required room |
| 11 | KingRoomFactory | Builds king rooms |
| 12 | SuiteRoomFactory | Builds suite rooms |
| 13 | SingleRoomFactory | Build single rooms |
| 14 | TwinBedRoomFactory | Builds twin bed rooms |

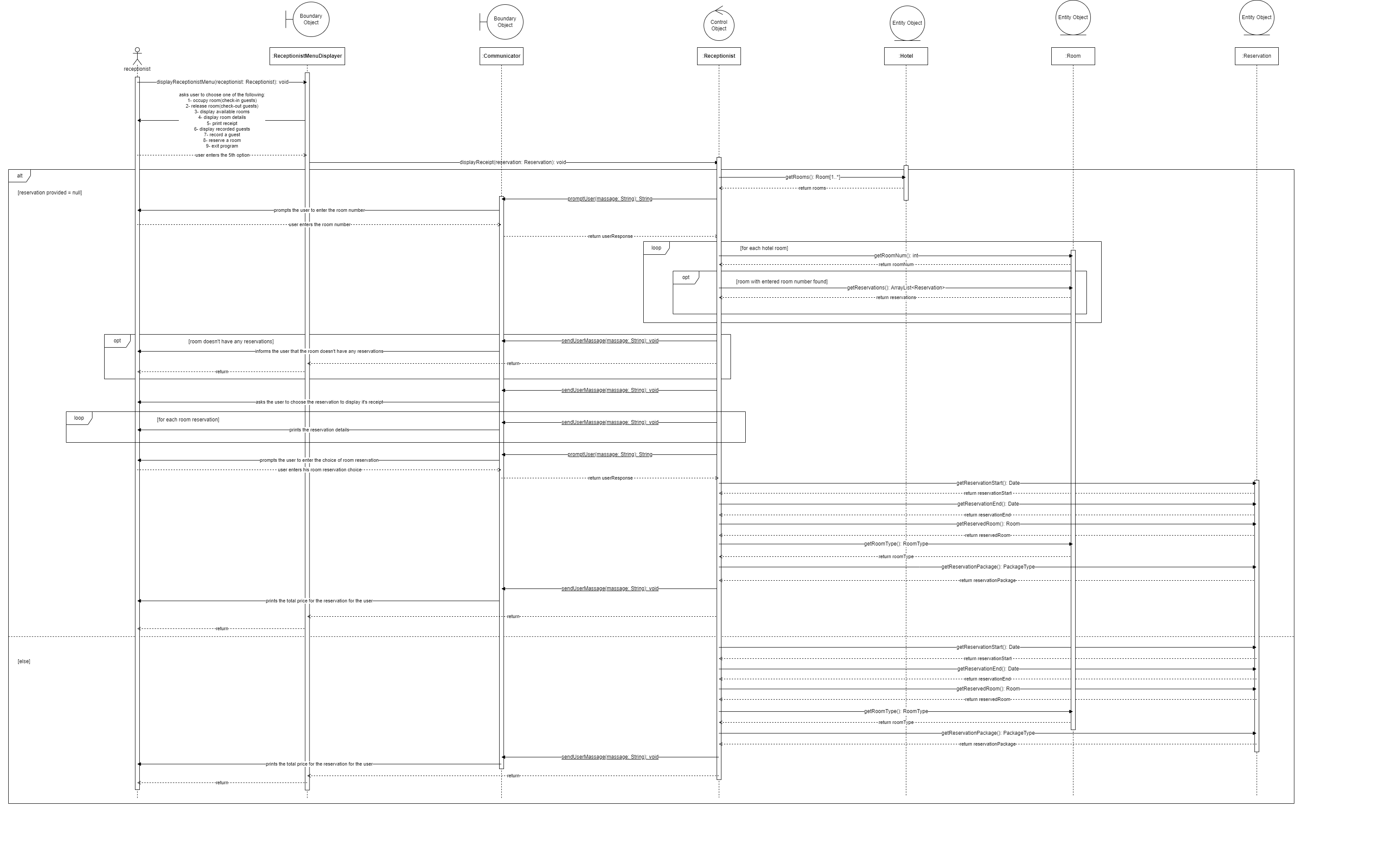
### Important Algorithm

## II. Sequence diagrams

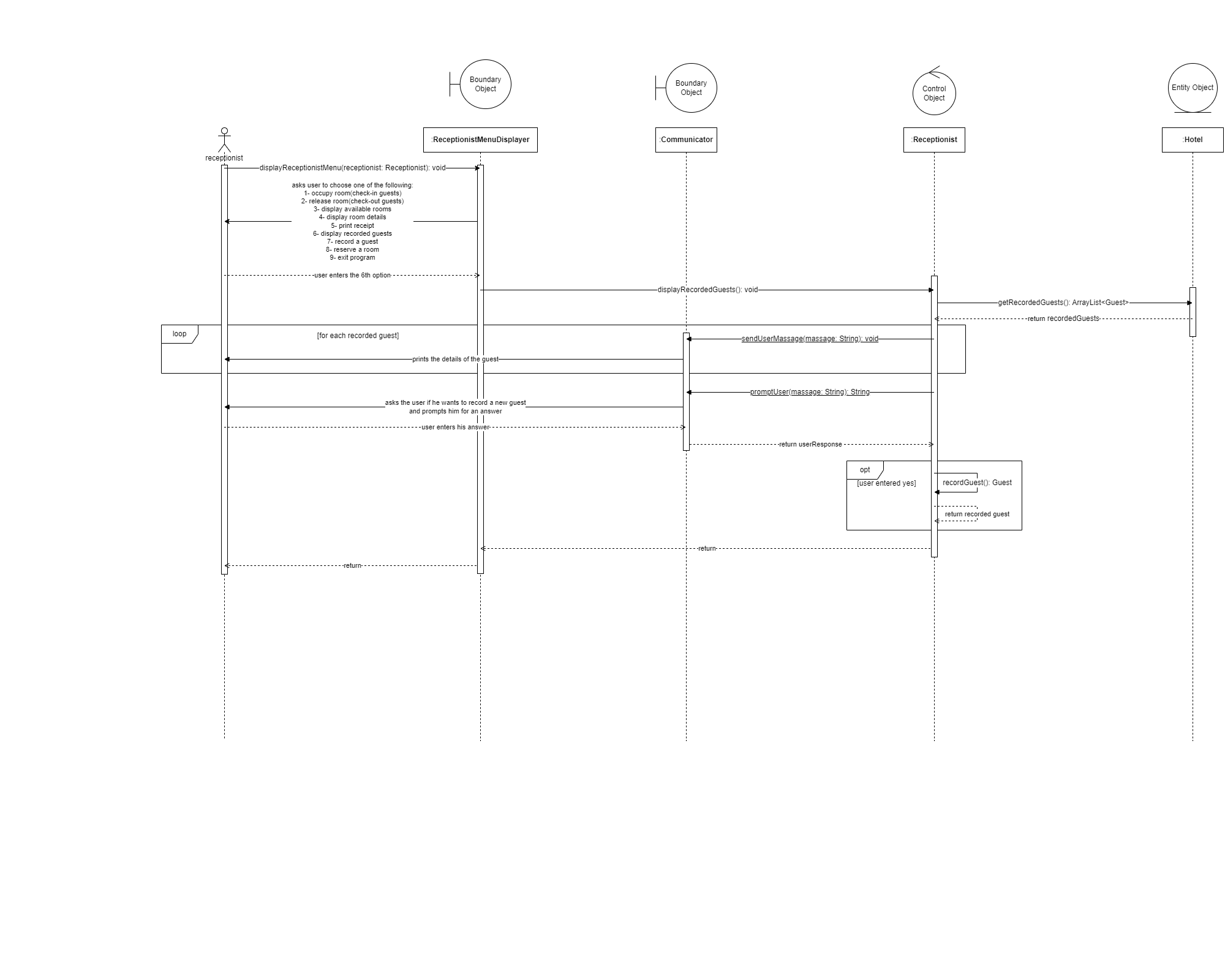
Use case 8 (display available rooms)



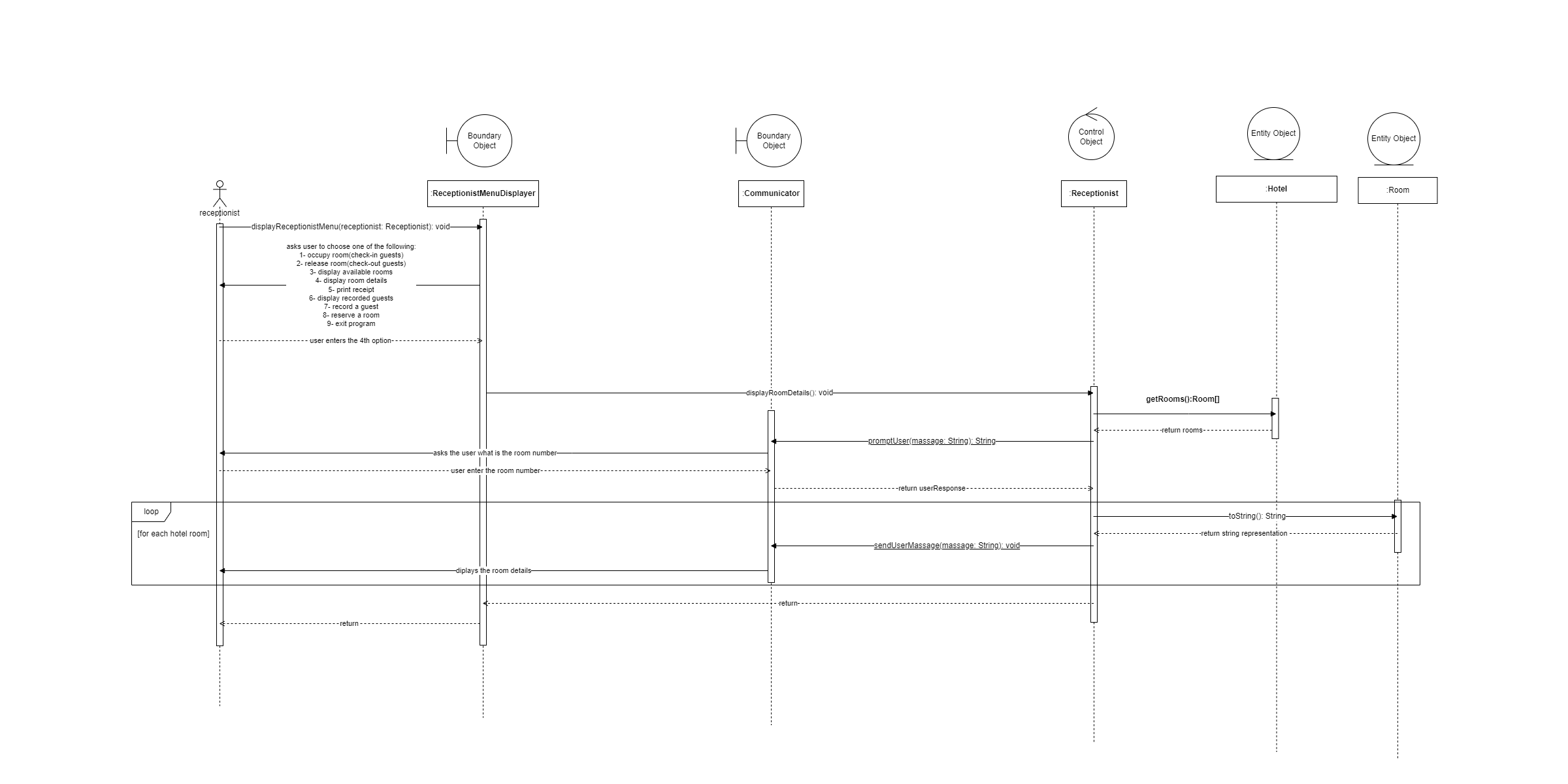
Use case 10(Print receipt)



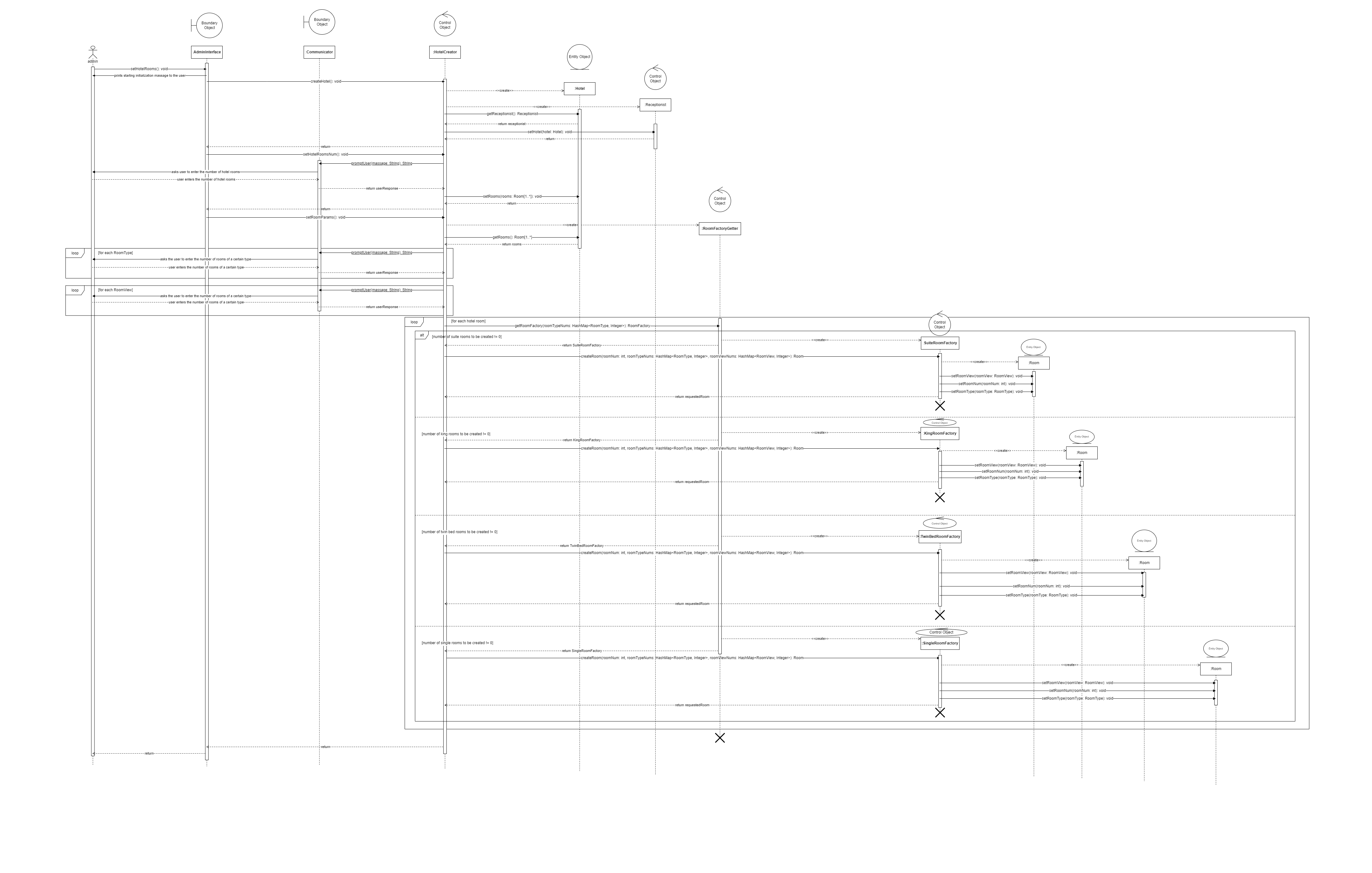
Use case 4(display recorded guests)



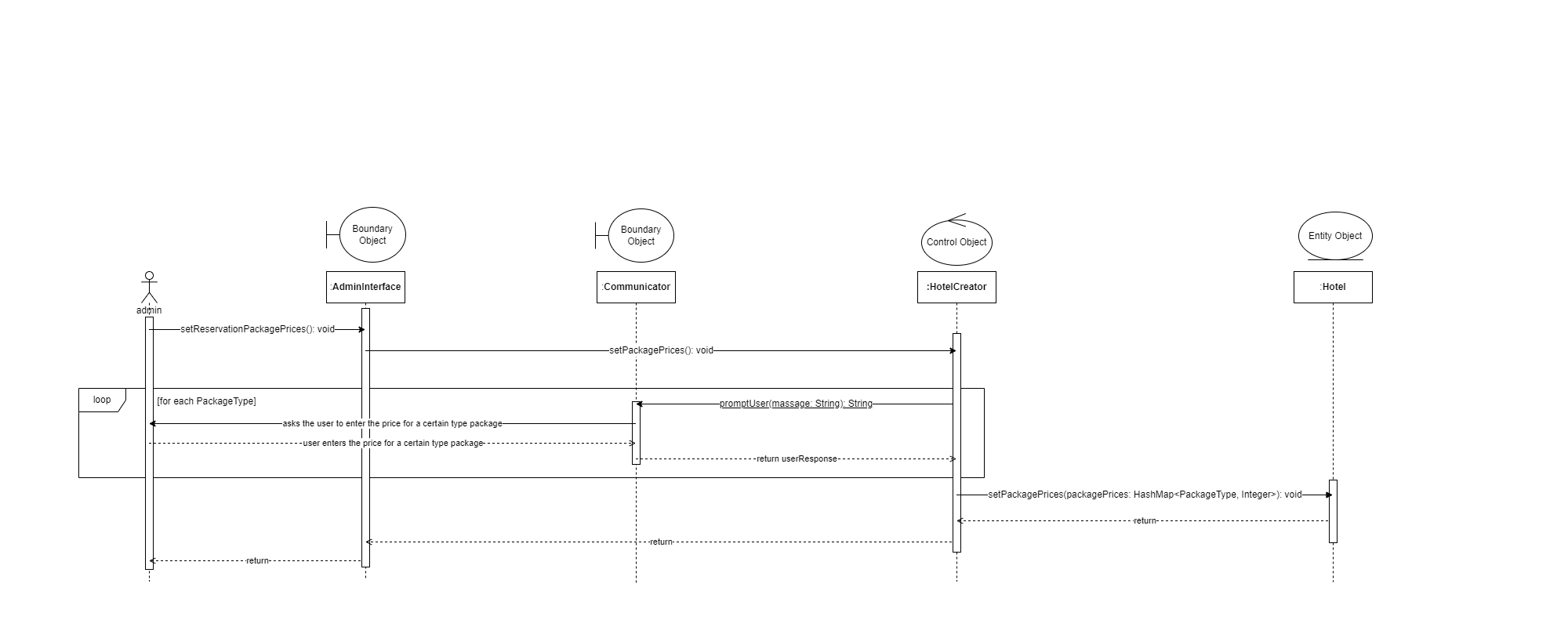
Use case 9(display a room details)



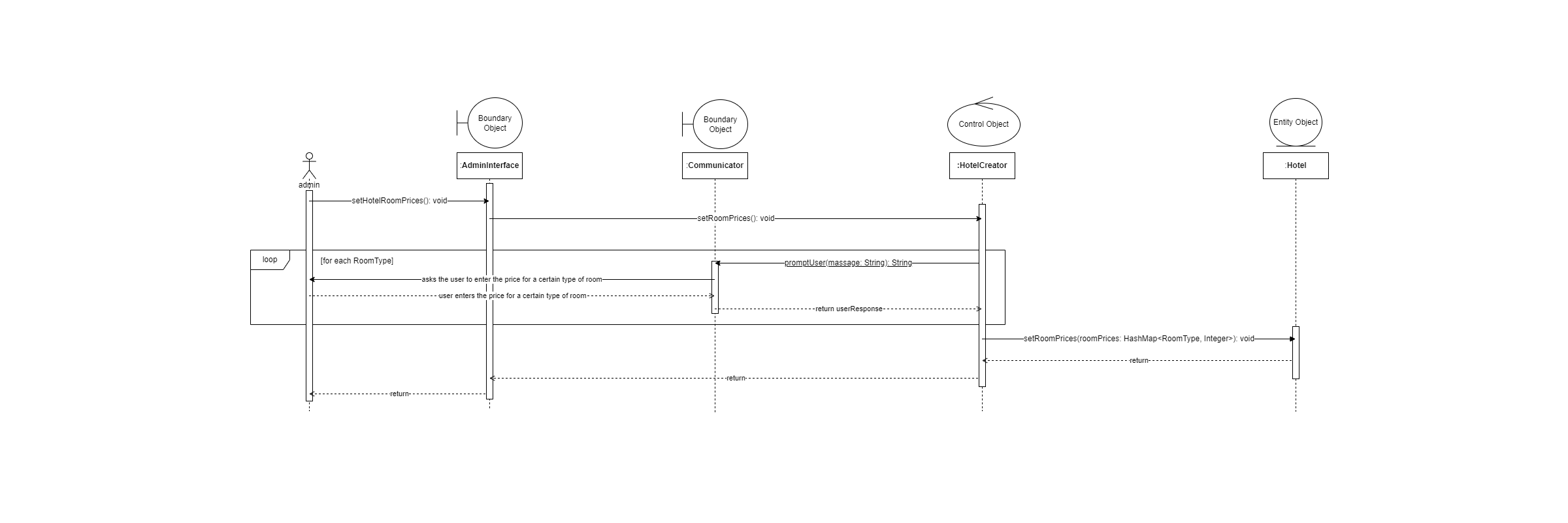
Use case 7(initialize Number of hotel rooms and their type and view)



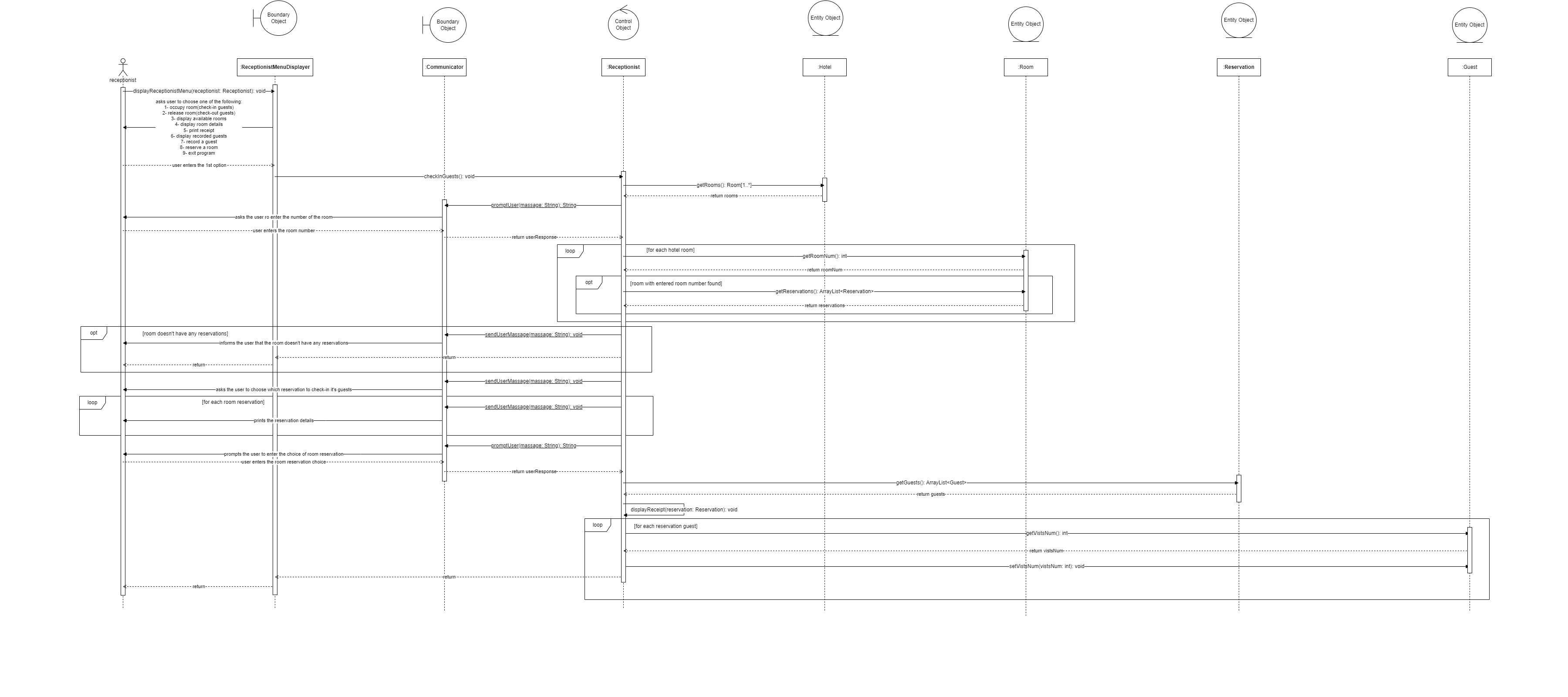
Use case 3(Initialize the price for each type of package)



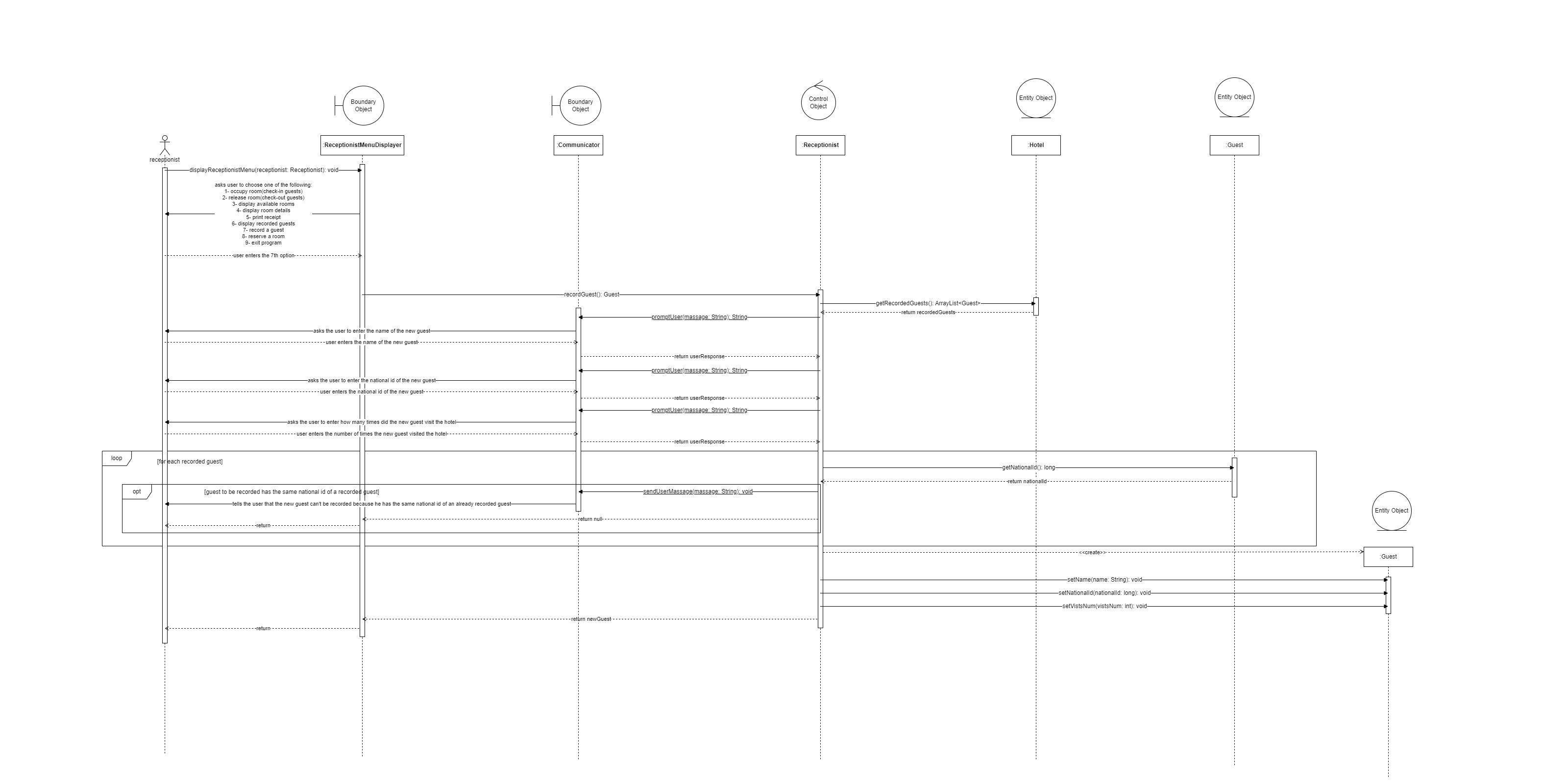
Use case 6(Initialize the price for each type of room)



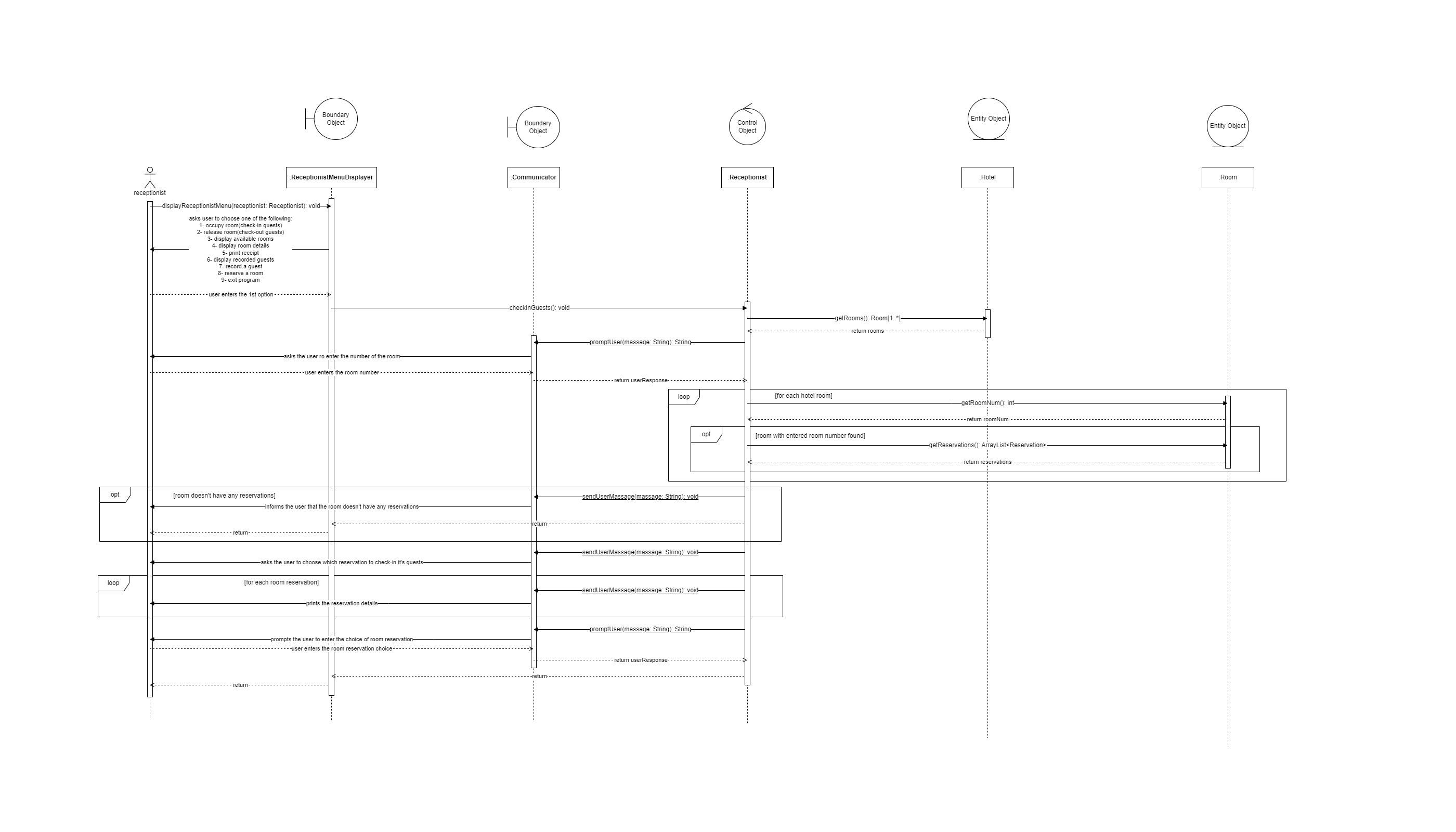
Use case2(Occupy rooms)



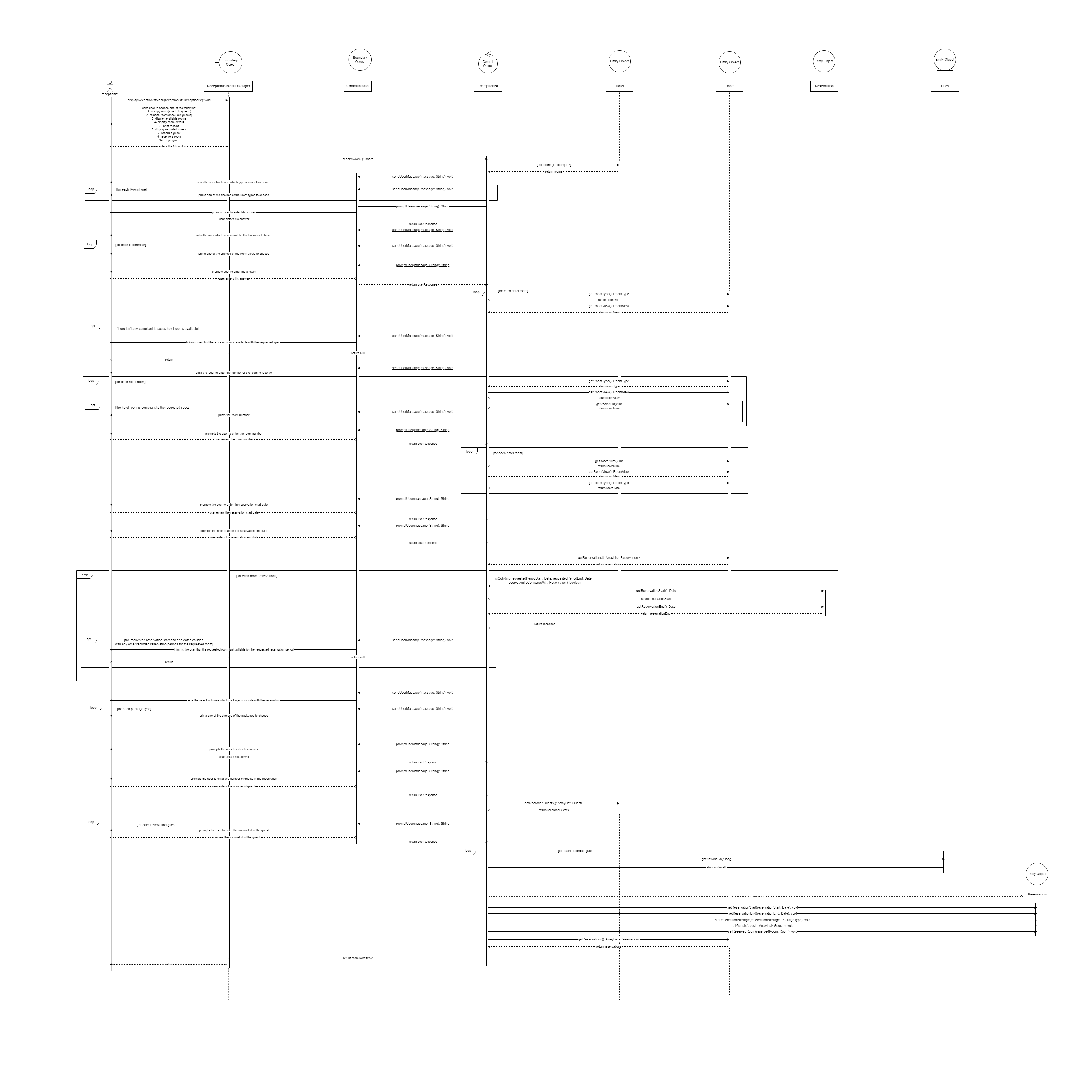
Use case 5(Record a new guest)



Use case 1(Release a room)



Use case 11(Reserve a room)



### Class - Sequence Usage Table

| **Class Name** | **Sequence Diagrams** | **Overall used methods** |
| --- | --- | --- |
| ReceptionistMenuDisplaye | 1,2,4,5,8,9,10,11 | + DisplayReceptionistMenu() |
| AdminInterface | 3,6,7 | + setHotelRooms(): void  + setHotelRoomPrices(): void  + setReservationPackagePrices(): void  + getResult(): Receptionist |
| HotelCreator | 3,6,7 | + createHotel(): void  + setHotelRoomsNum(): void  + setRoomParams(): void  + setRoomPrices(): void  + setPackagePrices(): void  + getReceptionist(): Receptionist |
| KingRoomFactory | 7 | + createRoom(roomNum: int, roomTypeNums: HashMap, roomViewNums: HashMap): Room |
| SuiteRoomFactory | 7 | + createRoom(roomNum: int, roomTypeNums: HashMap, roomViewNums: HashMap): Room |
| SingleRoomFactory | 7 | + createRoom(roomNum: int, roomTypeNums: HashMap, roomViewNums: HashMap): Room |
| TwinBedRoomFactory | 7 | + createRoom(roomNum: int, roomTypeNums: HashMap, roomViewNums: HashMap): Room |
| RoomFactoryGetter | 7 | + getRoomFactory(roomTypeNums: HashMap): RoomFactory |
| Hotel | 1,2,3,4,5,6,7,8,9,10,11 | + getRooms(): Room[1..\*]  + getRecordedGuests(): ArrayList  + getReceptionist(): Receptionist  + getRoomPrices(): HashMap  + getPackagePrices(): HashMap  + setRooms(rooms: Room[1..\*]): void  + setRecordedGuests(recordedGuests: ArrayList): void  + setReceptionist(receptionist: Receptionist): void  + setRoomPrices(roomPrices: HashMap): void  + setPackagePrices(packagePrices: HashMap): void) |
| Room | 11,1,2,7,9,10,8 | + getRoomNum(): int  + getRoomView(): RoomView  + getRoomType(): RoomType  + getReservations(): ArrayList  + setRoomNum(roomNum: int): void  + setRoomView(roomView: RoomView): void  + setRoomType(roomType: RoomType): void  + setReservations(reservations: ArrayList): void  + toString(): String |
| Reservation | 11,1,2,10,8 | + getReservedRoom(); Room  + getReservationStart(): Date  + getReservationEnd(): Date  + getReservationPackage(): PackageType  + getGuests(): ArrayList  + setReservedRoom(reservedRoom: Room): void  + setReservationStart(reservationStart: Date): void  + setReservationEnd(reservationEnd: Date): void  + setReservationPackage(reservationPackage: PackageType): void  + setGuests(guests: ArrayList): void |
| Guest | 11,5,2,4 | + getName(): String  + getNationalId(): long  + getVistsNum(): int  + setName(name: String): void  + setNationalId(nationalId: long): void  + setVistsNum(vistsNum: int): void |
| Receptionist | 1,2,4,5,7,8,9,10,11 | + getHotel(): Hotel  + setHotel(hotel: Hotel): void  + displayRecordedGuests(): void  - isColliding(requestedPeriodStart: Date, requestedPeriodEnd: Date, reservationToCompareWith: Reservation): boolean  + reservRoom(): Room  + checkInGuests(): void  + checkOutGuests(): void  + recordGuest(): Guest  + displayAvailableRooms(): void  + displayRoomDetails(): void  + displayReceipt(reservation: Reservation): void |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Did all the class diagram and all the sequence diagrams except for use cases 8,9 | *Omar Mohamed Elzohry* |
| Did sequence diagram 8,9 and updated by omar | *Ahmed mohamed* |