Hotel Reservation System

Having an efficient reservation system is essential for the business of a hotel. To achieve this goal, we decide to develop a hotel reservation system. This platform will be developed using Java.

Major functionalities

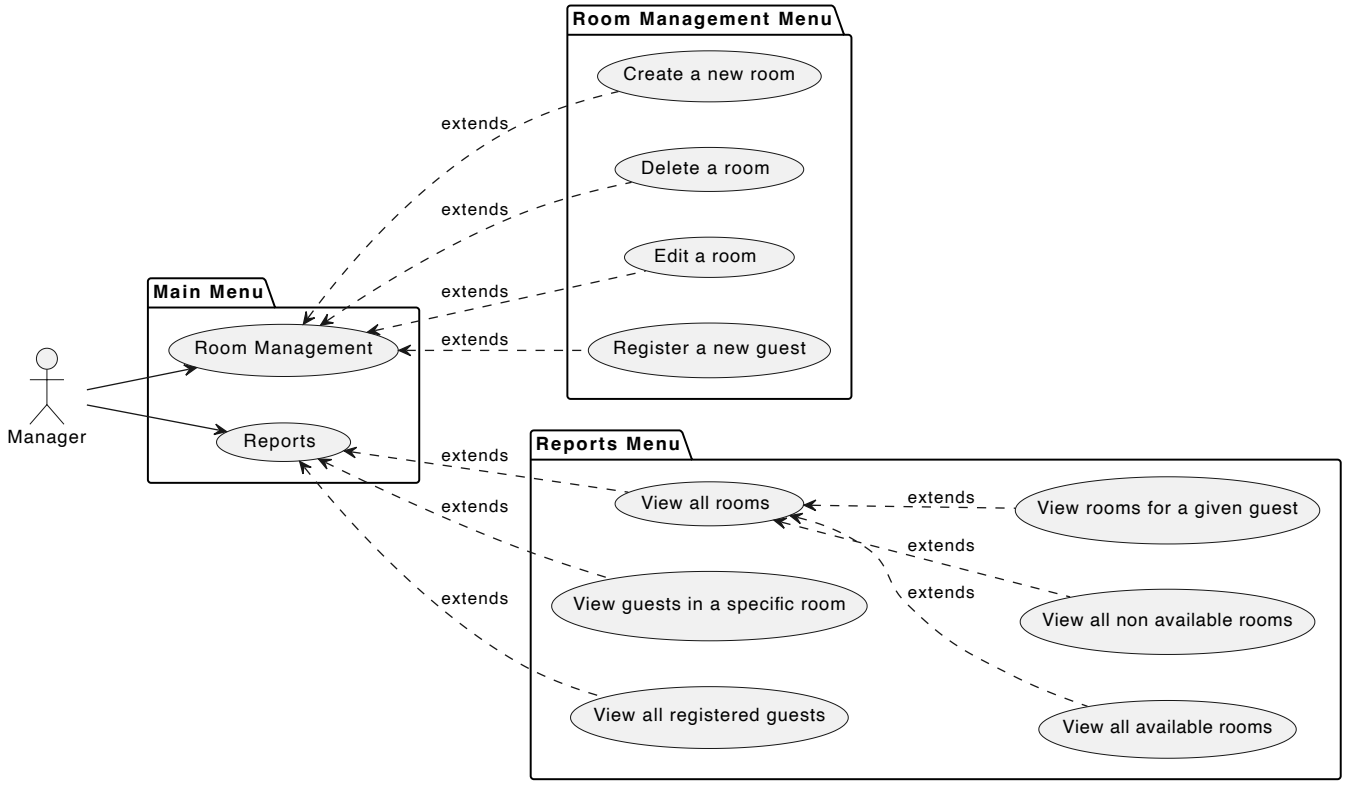
Our system is designed for two kinds of users: guests and managers. A guest can use this application to browse rooms' information and show your booking or leaving intention to the hotel manager. At the same time, a hotel manager can also use this system, including creating rooms, deleting rooms, editing rooms, browsing room information, checking room availability, adding, removing a guest, etc.

We prepared accounts

Role	Username	Password
Manager	manager	manager001
Guest	user	user001

Use Case Diagram

Manager Use Case

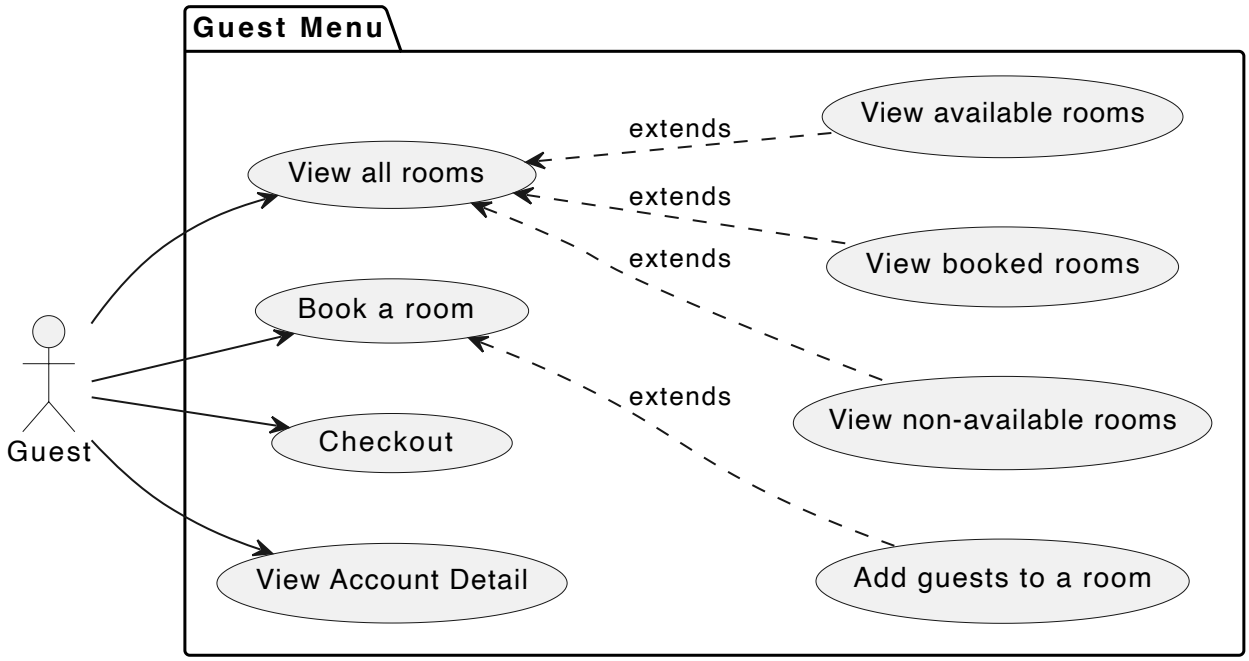


UC REFERENCE	USECASE NAME	OVERVIEW	RELATED USE CASES	ACTORS	PRE CONDIT
UC1	Room Management	Access management functions related to room operations.	UC2, UC3, UC4, UC5	Manager	Manager must be logged in
UC2	Create a new room	Allows the manager to add a new room to the hotel.	UC1	Manager	Manager the Room Manager menu.
UC3	Delete a room	Permits the manager to remove an existing room	UC1	Manager	Manager selects a room to delete.

UC REFERENCE	USECASE NAME	OVERVIEW	RELATED USE CASES	ACTORS	PRE CONDIT
		from the hotel.			
UC4	Edit a room	Enables the manager to modify the details of an existing room.	UC1	Manager	A room to is selecte
UC5	Register a new guest	Allows the manager to add a new guest to the system.	UC1	Manager	Manager the Room Manager menu.
UC6	Reports	Access reports related to room and guest management.	UC7, UC8, UC9, UC10, UC11	Manager	Manager selects th Reports menu.
UC7	View all rooms	Manager can see all rooms, irrespective of their status.	UC6, UC8, UC9	Manager	Manager the Repo menu.
UC8	View all available rooms	Displays all rooms currently	UC7	Manager	Part of viewing a rooms.

UC REFERENCE	USECASE NAME	OVERVIEW	RELATED USE CASES	ACTORS	PRE CONDIT
		available for booking.			
UC9	View all non-available rooms	Displays rooms that are currently booked or unavailable.	UC7	Manager	Part of viewing a rooms.
UC10	View guests in a specific room	Shows details of guests currently staying in a specific room.	UC6, UC7	Manager	Manager selects a specific r
UC11	View rooms for a given guest	Shows all rooms associated with a specific guest.	UC6, UC7	Manager	Manager selects a guest.
UC12	View all registered guests	Shows a list of all guests registered in the hotel.	UC6, UC7	Manager	Manager the Repo menu.

Guest Use Case



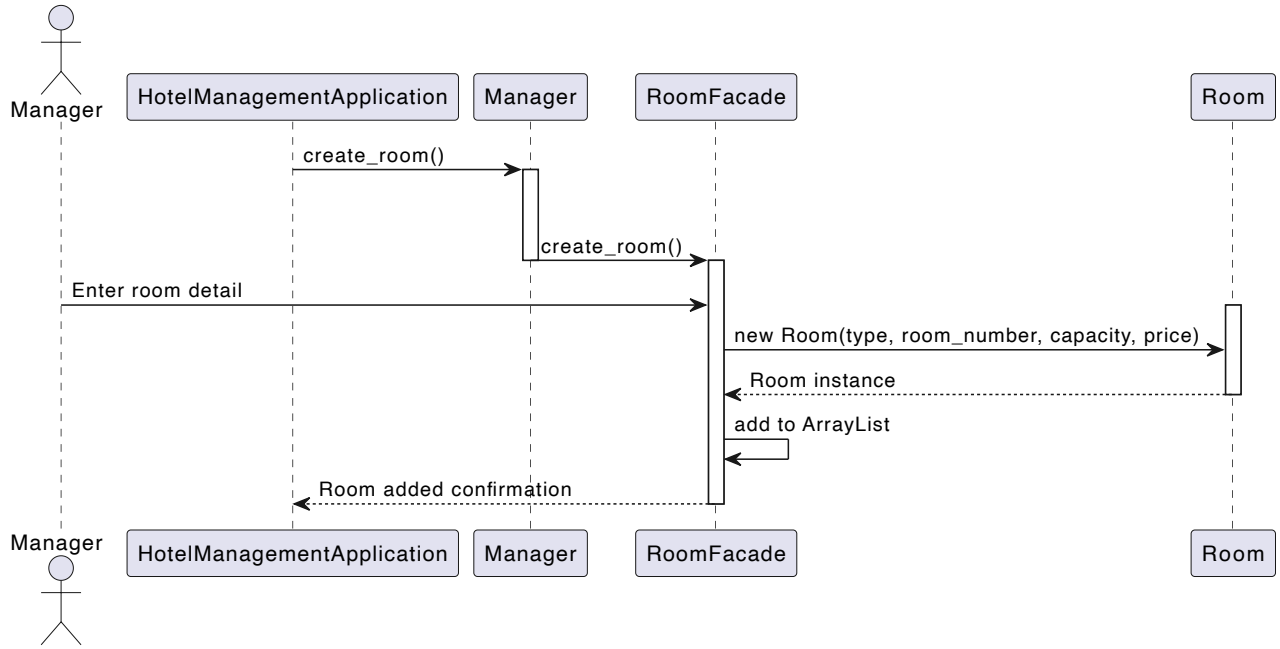
UC REFERENCE	USECASE NAME	OVERVIEW	RELATED USE CASES	ACTORS	PRE CONDITION
UC1	View all rooms	Guest can view all rooms, regardless of availability.	UC2, UC3, UC5	Guest	Guest is logged in.
UC2	View available rooms	Guest views rooms that are currently available for booking.	UC1	Guest	Guest selects to view rooms.
UC3	View non-available rooms	Guest sees rooms that are currently	UC1	Guest	Guest selects to view rooms.

UC REFERENCE	USECASE NAME	OVERVIEW	RELATED USE CASES	ACTORS	PRE CONDITION
		not available.			
UC4	Book a room	Guest books a room for a specific date.	None	Guest	Guest selects a room to book.
UC5	View booked rooms	Guest views all rooms they have booked.	UC1	Guest	Guest is logged in.
UC6	Add guests to a room	Guest can add more guests to their room booking.	UC4	Guest	Guest has a booked room
UC7	Checkout	Guest checks out of a room, ending their booking.	None	Guest	Guest decides to end stay.
UC8	View Account Detail	Guest views their account details and booking history.	None	Guest	Guest accesses their account

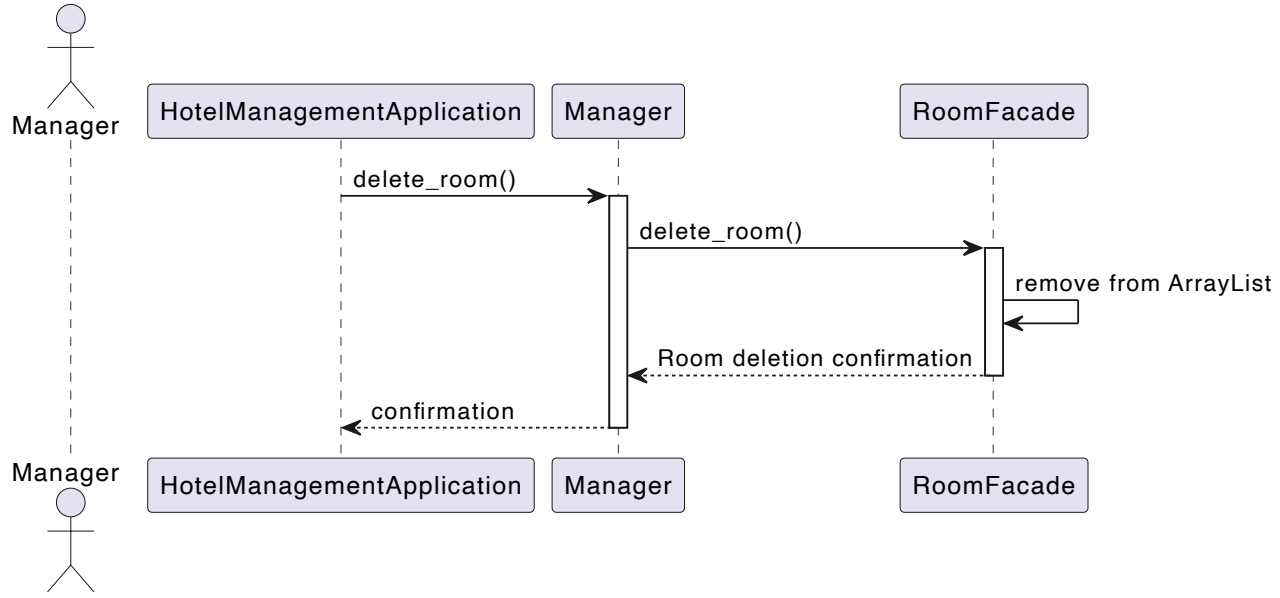
Sequence Diagram

Manager's sequence diagram

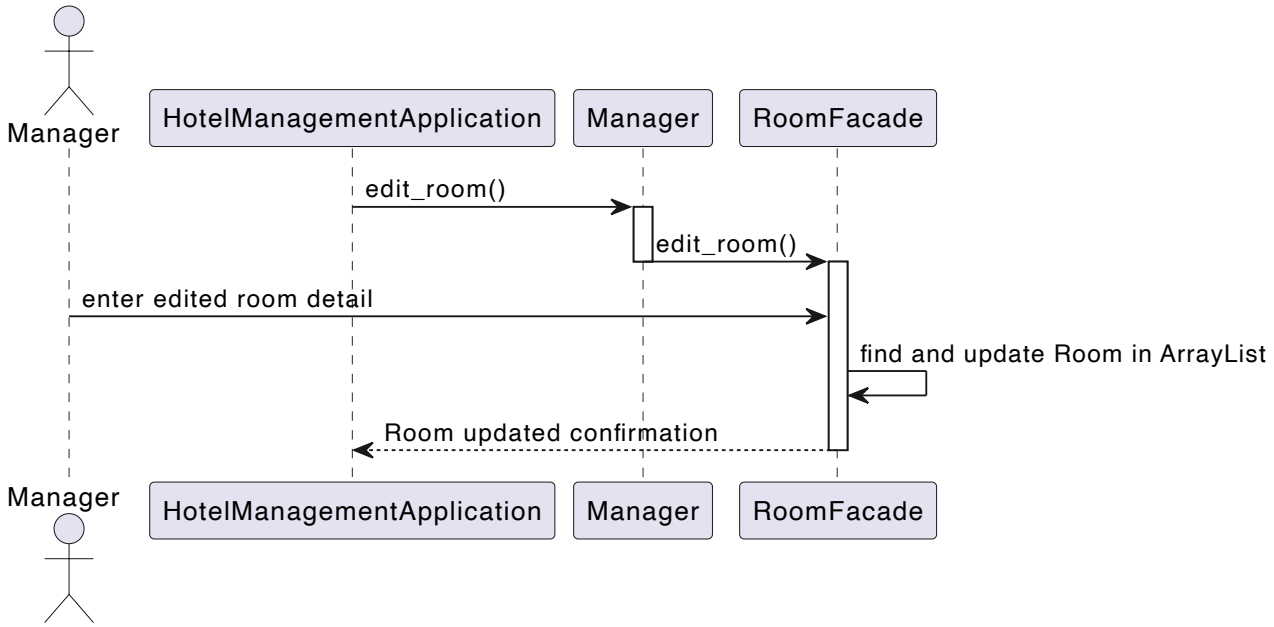
- create a room



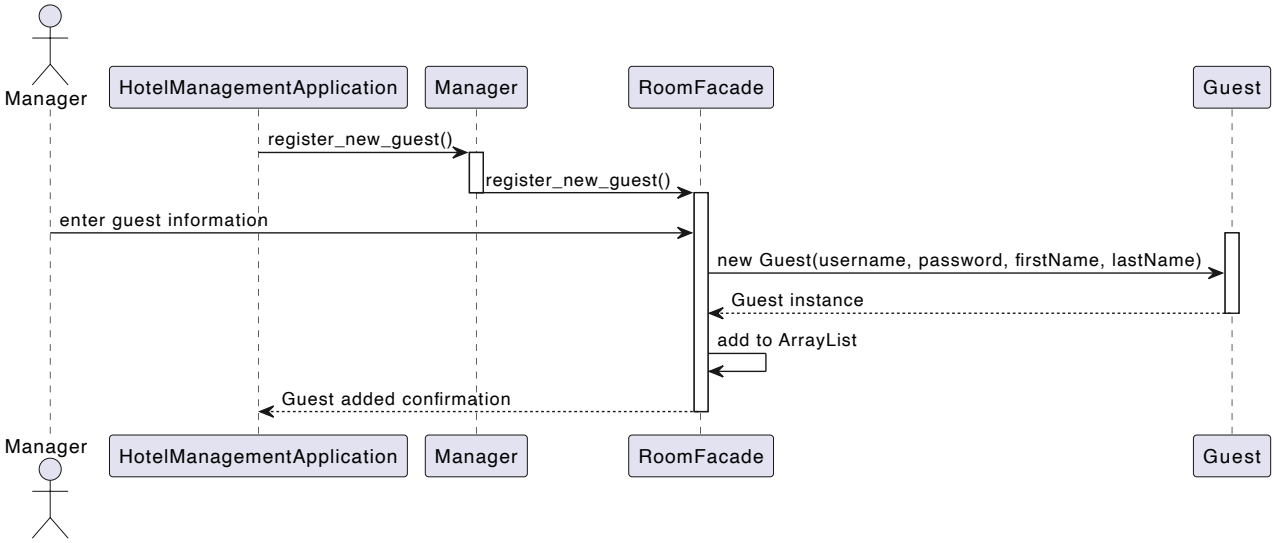
- delete a room



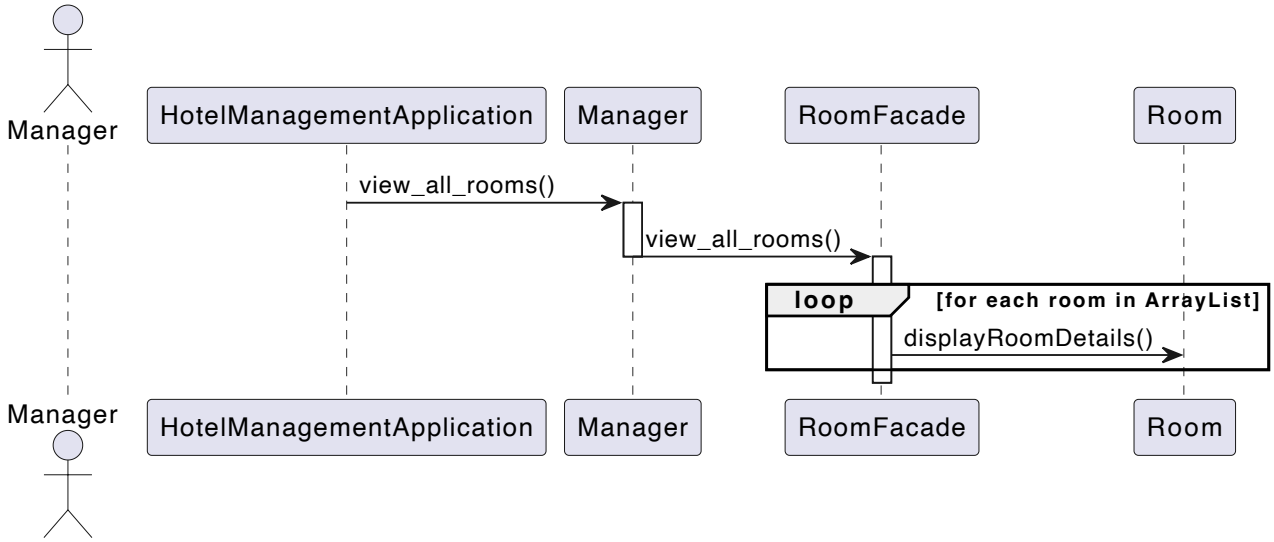
- edit a room



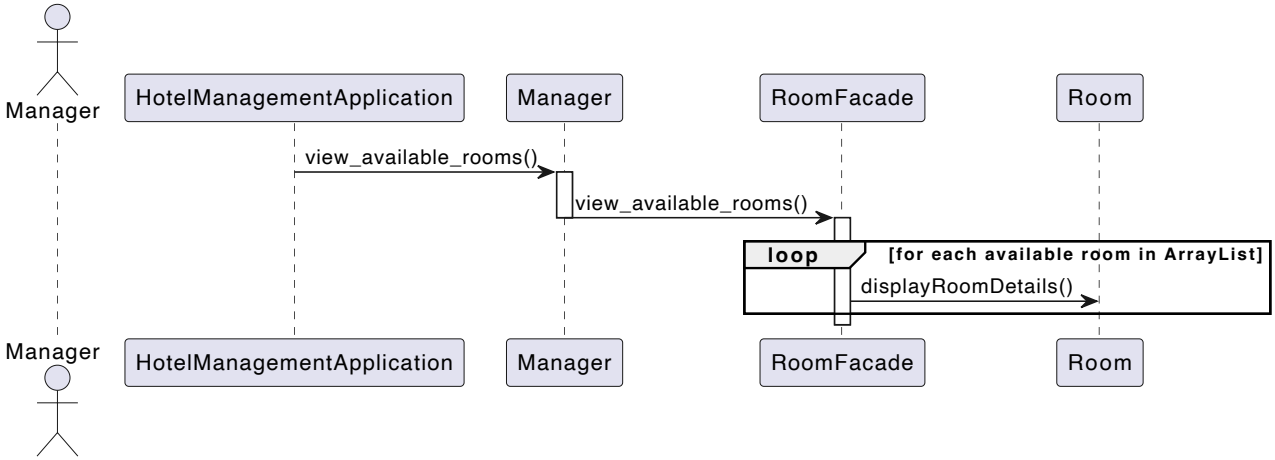
- register a new guest



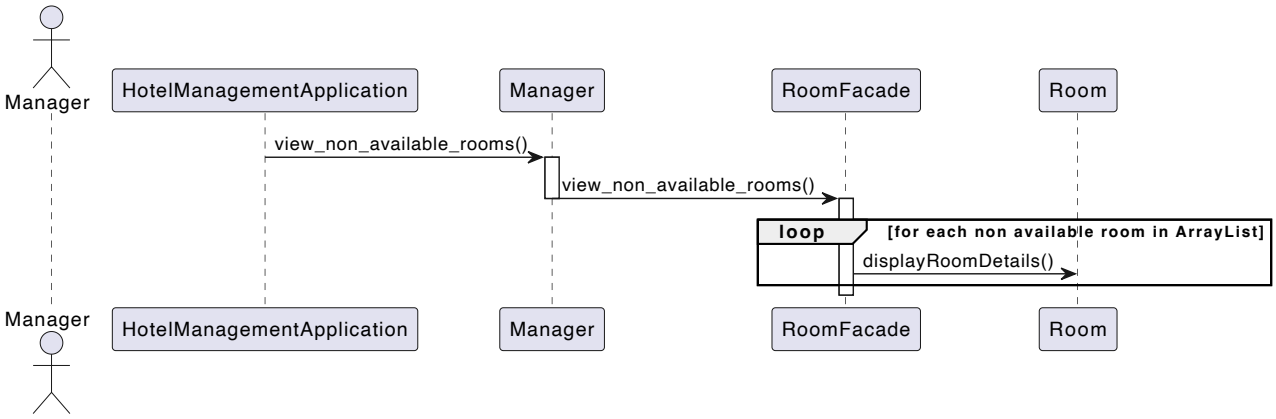
- view all rooms



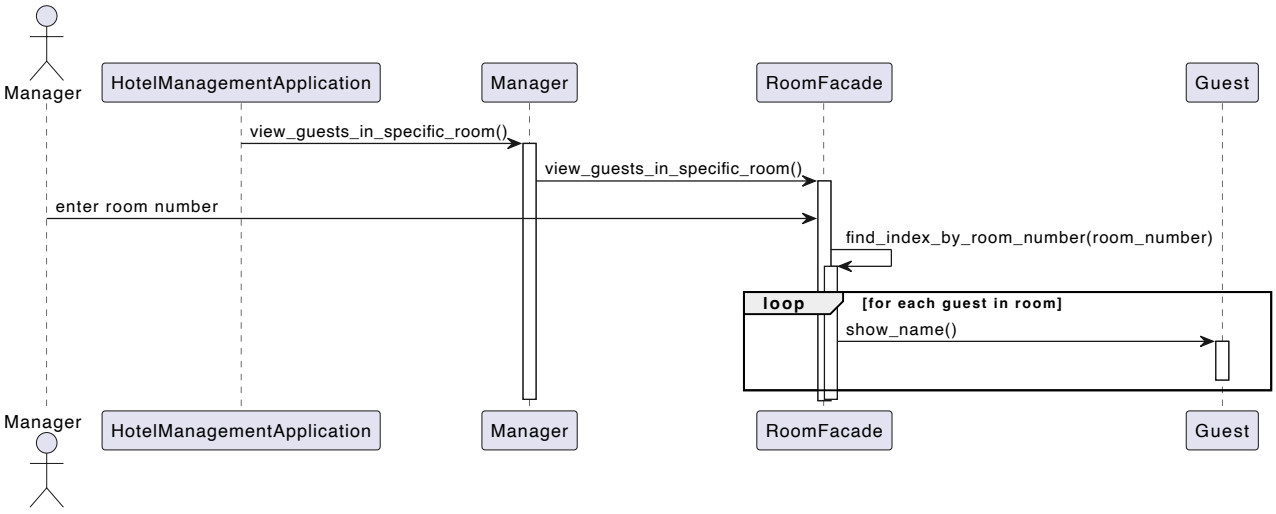
- view all available rooms



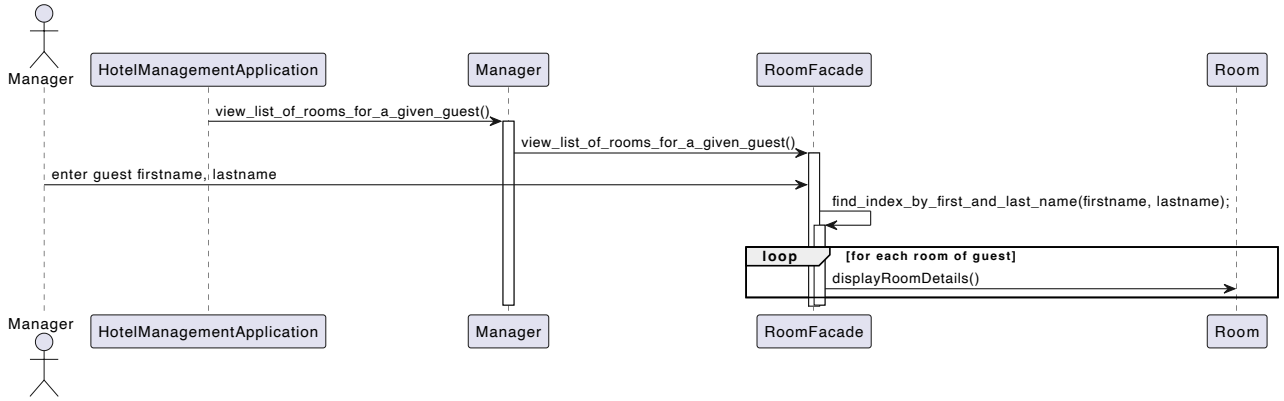
- view non all available rooms



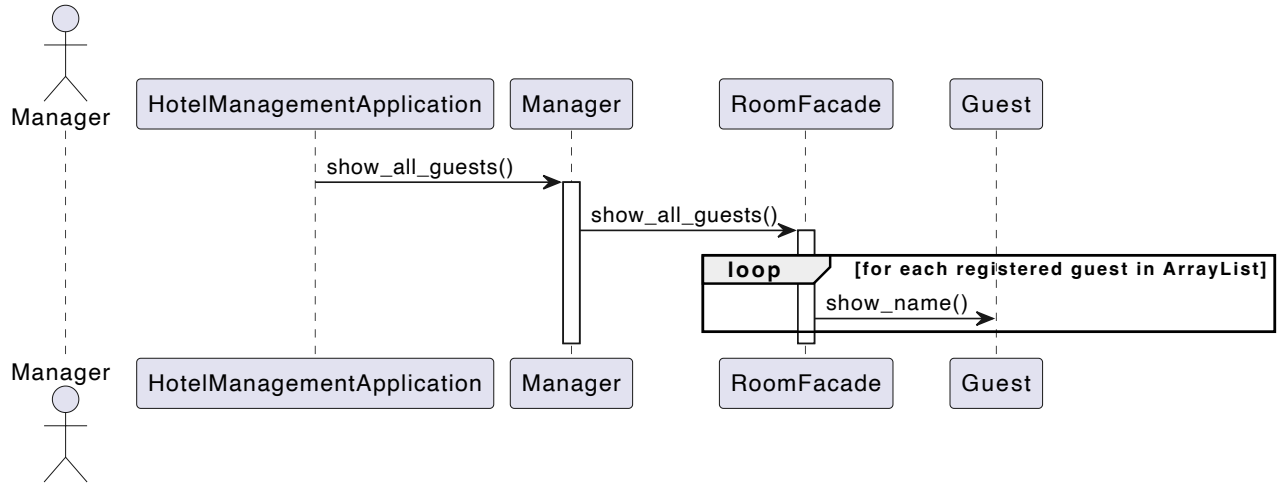
- View guests in a specific room



- View rooms for a given guest

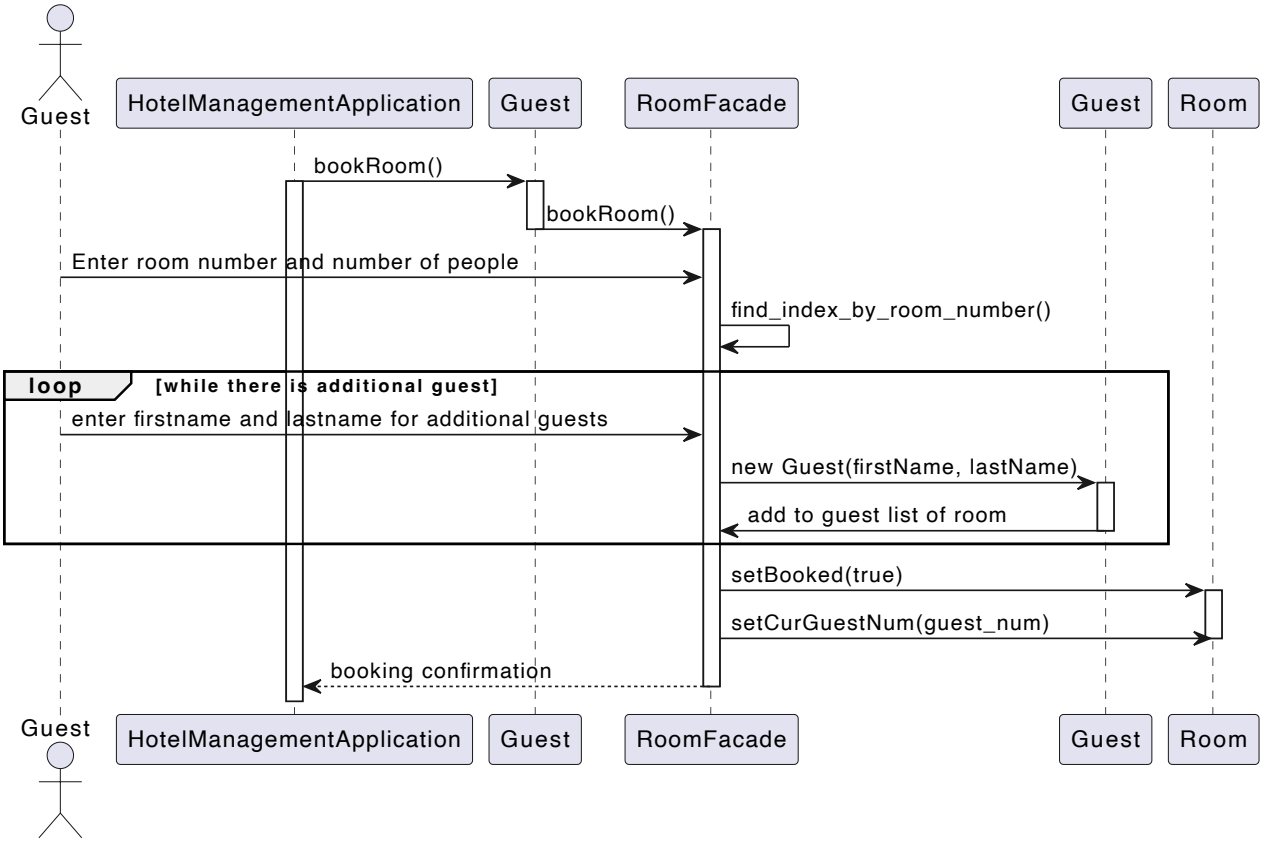


- View all registered guests

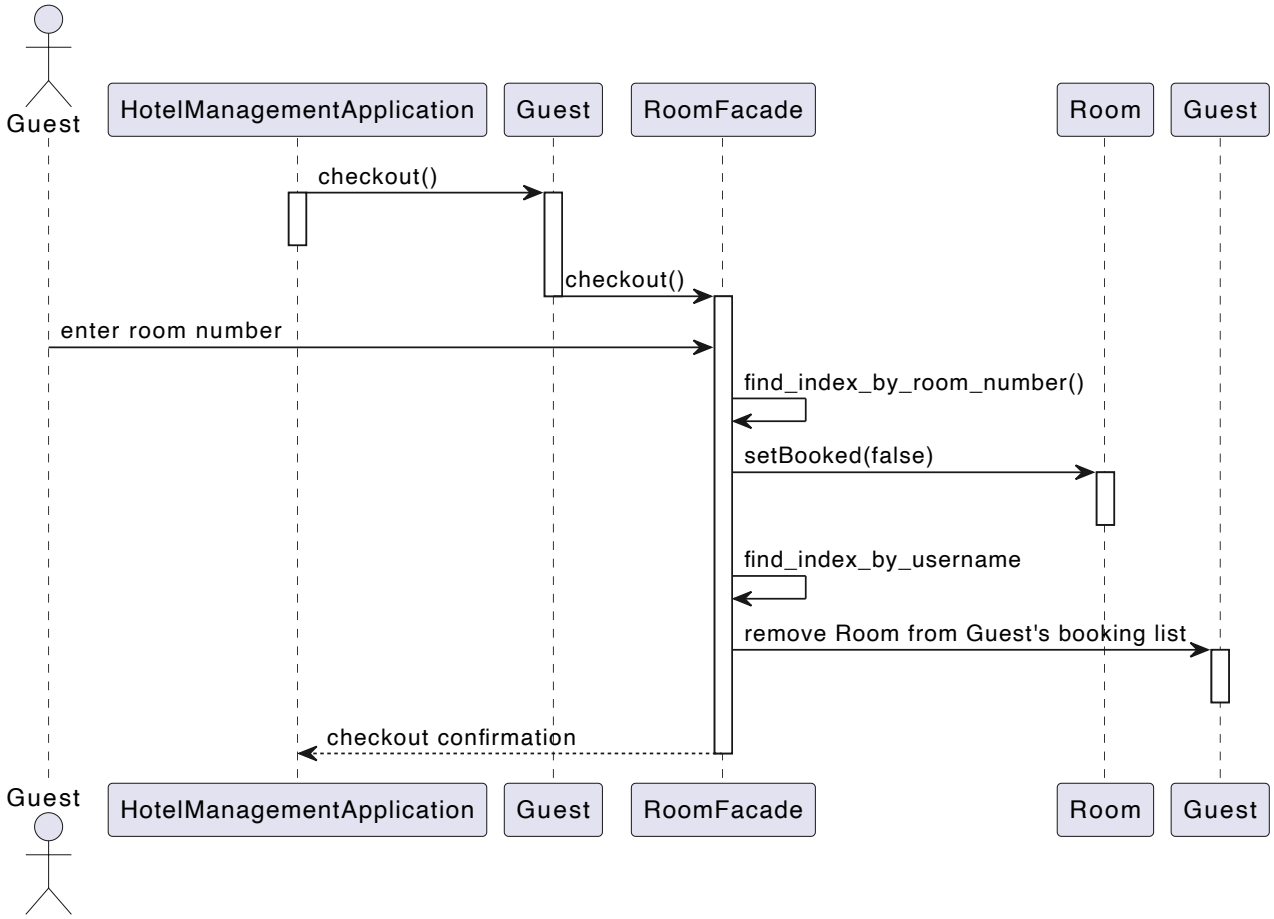


Guest's sequence diagram

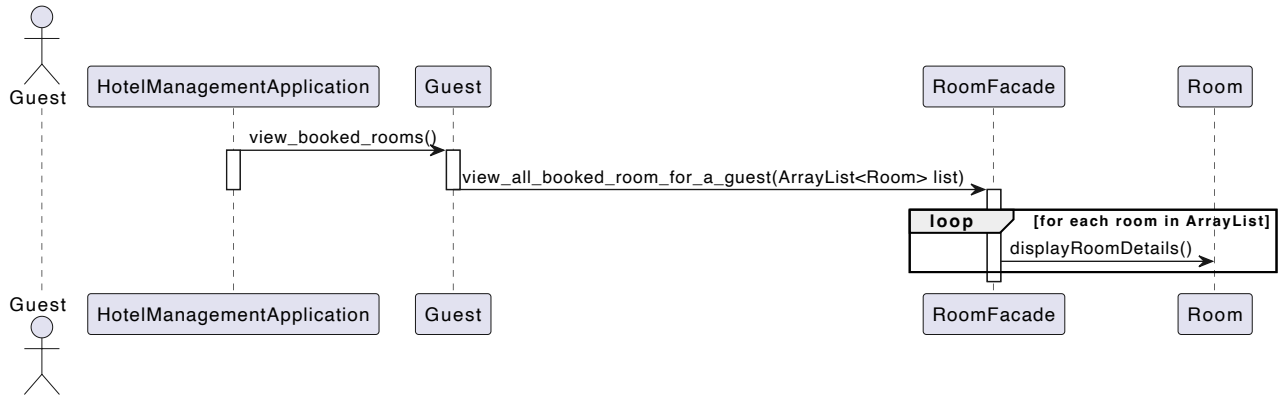
- book a room



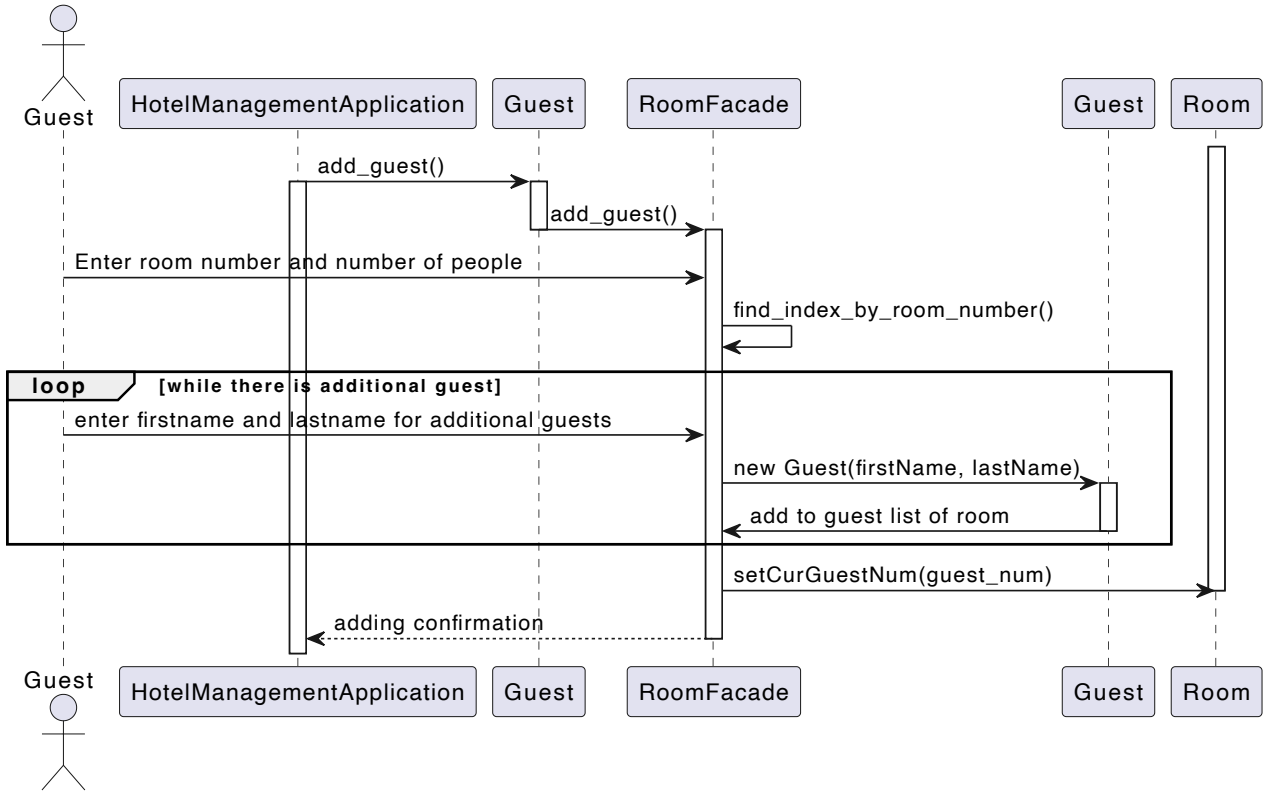
- checkout



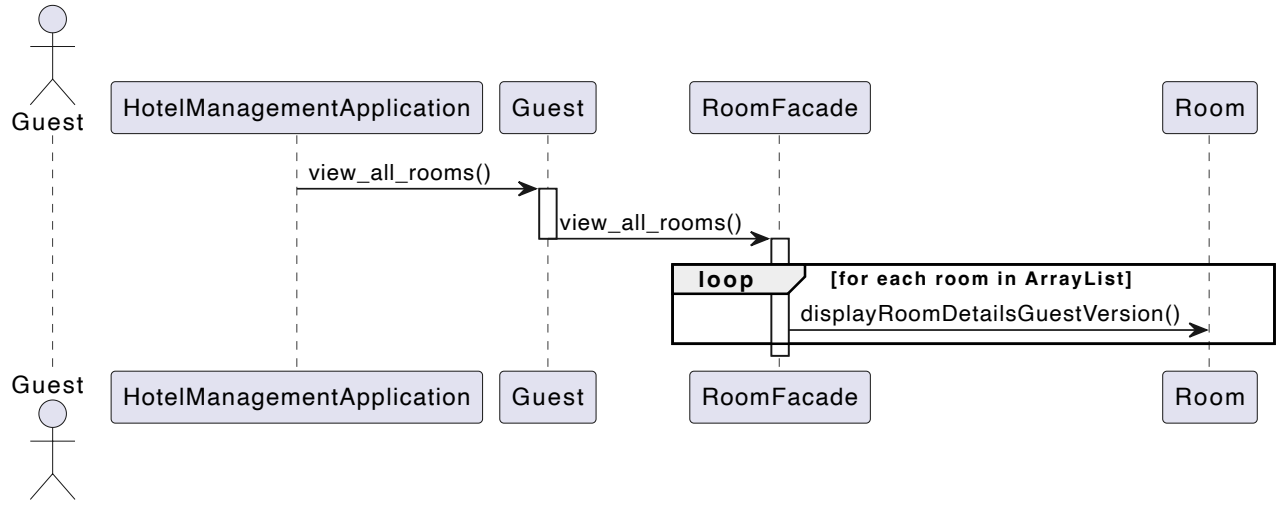
- view booked rooms



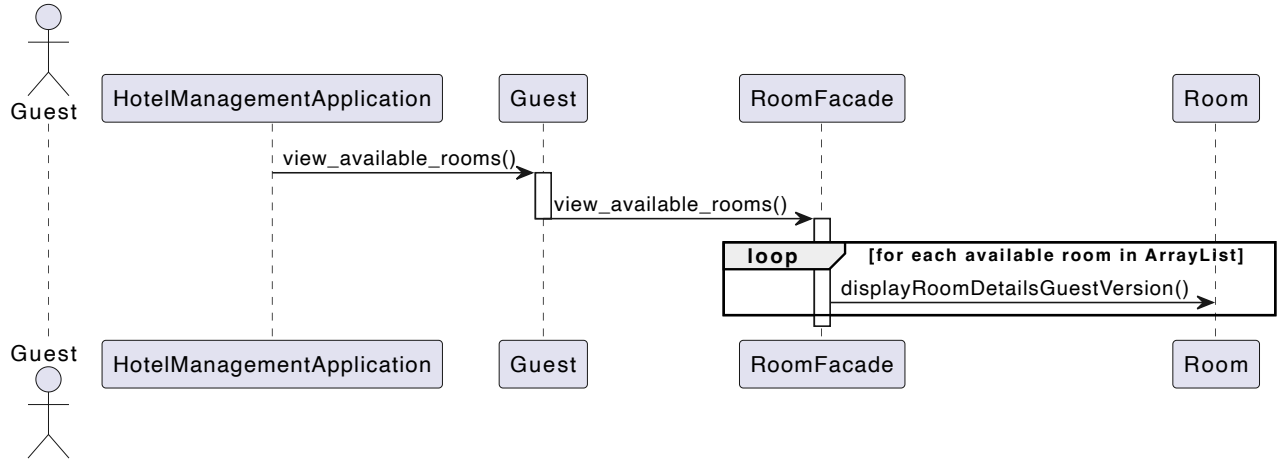
- add guests to a room



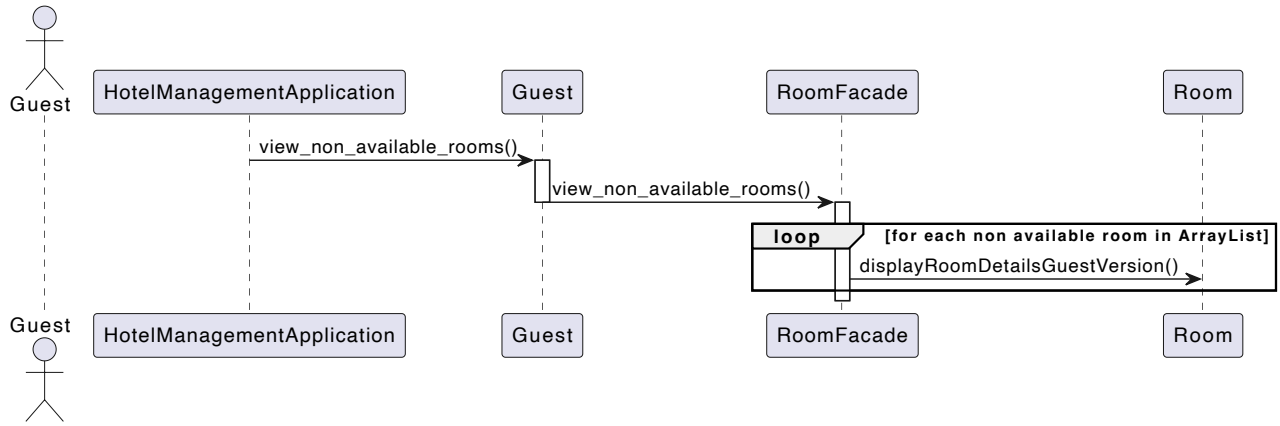
- view all rooms



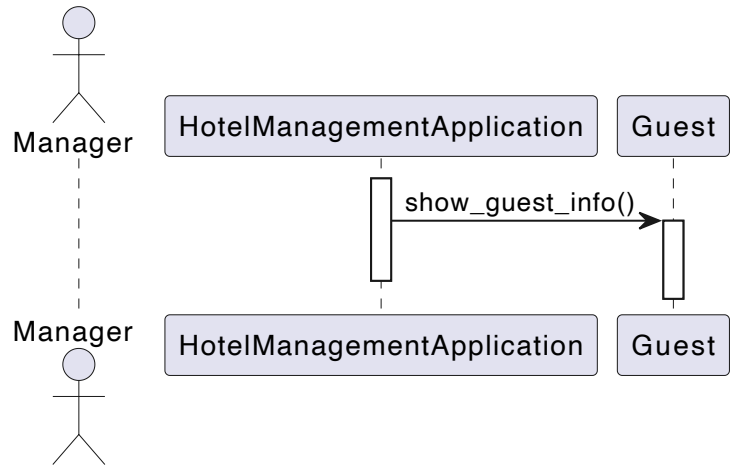
- view all available rooms



- view non all available rooms

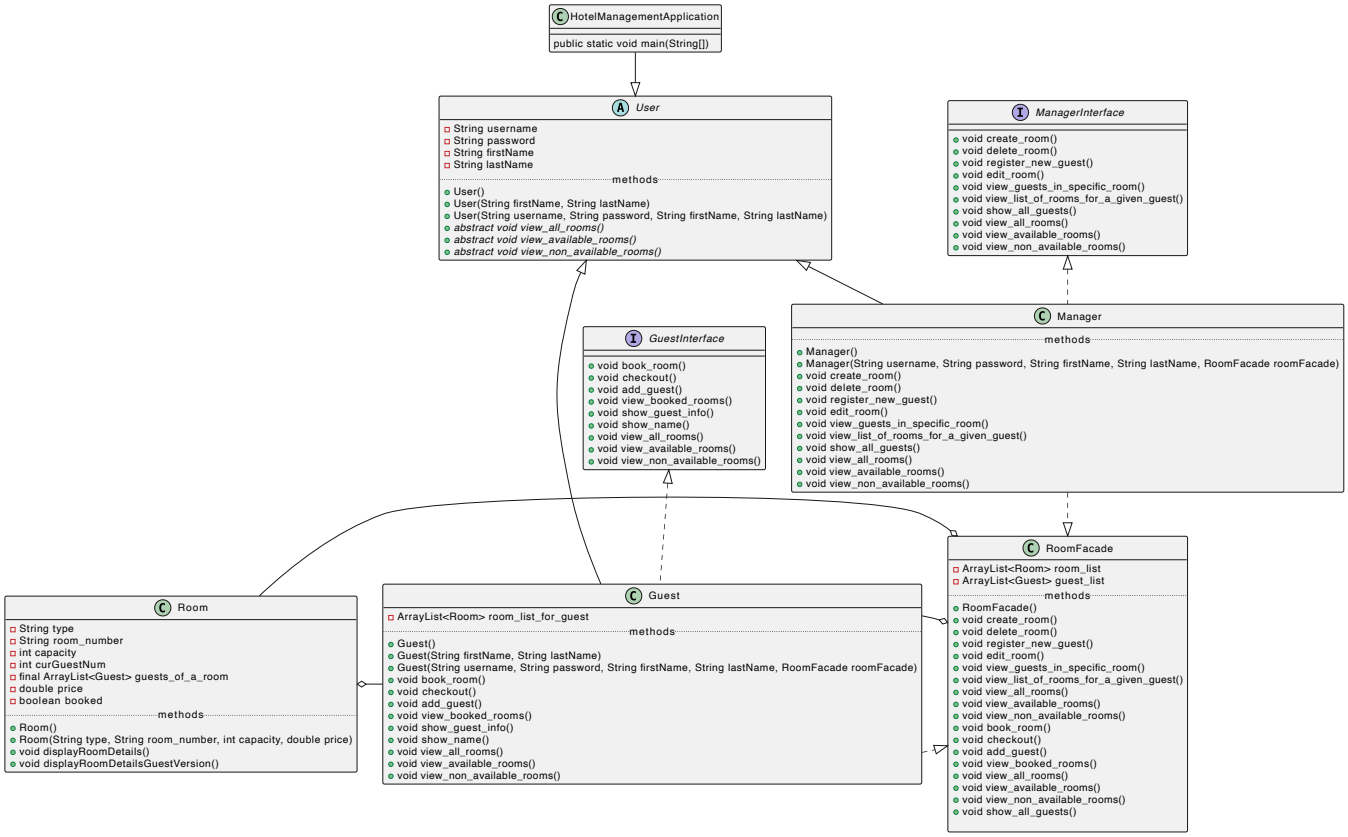


- view account detail



-

Class Diagram


HotelManagementApplication
public static void main(String[])
User
- String username
- String password
- String firstName
- String lastName
methods
+ User()
+ User(String firstName, String lastName)
+ User(String username, String password, String firstName, String lastName)
+ abstract void view_all_rooms()
+ abstract void view_available_rooms()
+ abstract void view_non_available_rooms()
ManagerInterface
+ void create_room()
+ void delete_room()
+ void register_new_guest()
+ void edit_room()
+ void view_guests_in_specific_room()
+ void view_list_of_rooms_for_a_given_guest()
+ void show_all_guests()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
GuestInterface
+ void book_room()
+ void checkout()
+ void add_guest()
+ void view_booked_rooms()
+ void show_guest_info()
+ void show_name()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
Manager
methods
+ Manager()
+ Manager(String username, String password, String firstName, String lastName, RoomFacade roomFacade)
+ void create_room()
+ void delete_room()
+ void register_new_guest()
+ void edit_room()
+ void view_guests_in_specific_room()
+ void view_list_of_rooms_for_a_given_guest()
+ void show_all_guests()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
Room
- String type
- String room_number
- int capacity
- int curGuestNum
- final ArrayList<Guest> guests_of_a_room
- double price
- boolean booked
methods
+ Room()
+ Room(String type, String room_number, int capacity, double price)
+ void displayRoomDetails()
+ void displayRoomDetailsGuestVersion()
Guest
- ArrayList<Room> room_list_for_guest
methods
+ Guest()
+ Guest(String firstName, String lastName)
+ Guest(String username, String password, String firstName, String lastName, RoomFacade roomFacade)
+ void book_room()
+ void checkout()
+ void add_guest()
+ void view_booked_rooms()
+ void show_guest_info()
+ void show_name()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
RoomFacade
- ArrayList<Room> room_list
- ArrayList<Guest> guest_list
methods
+ RoomFacade()
+ void create_room()
+ void delete_room()
+ void register_new_guest()
+ void edit_room()
+ void view_guests_in_specific_room()
+ void view_list_of_rooms_for_a_given_guest()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
+ void book_room()
+ void checkout()
+ void add_guest()
+ void view_booked_rooms()
+ void view_all_rooms()
+ void view_available_rooms()
+ void view_non_available_rooms()
+ void show_all_guests()