Functional Requirements:

Critical Priority:
1. As an employee, I want to make a reservation for a given interval, guest, start date, end date.
2. As an employee, I want to search for rooms by room type, smoking availability.
3. As an employee, I want to register a guest with: name, address, phone, nationality, check-in and check-out.
4. As an employee, I want to add, modify, display or delete reservations.
5. As an employee, I want to add, display or delete data about guests and rooms.

High Priority:
6. As an employee, I want to modify the price of rooms.
7. As an employee, I want to display the total amount to be paid by a guest during their stay.
8. As an employee, I want to check room availability on a specific date.
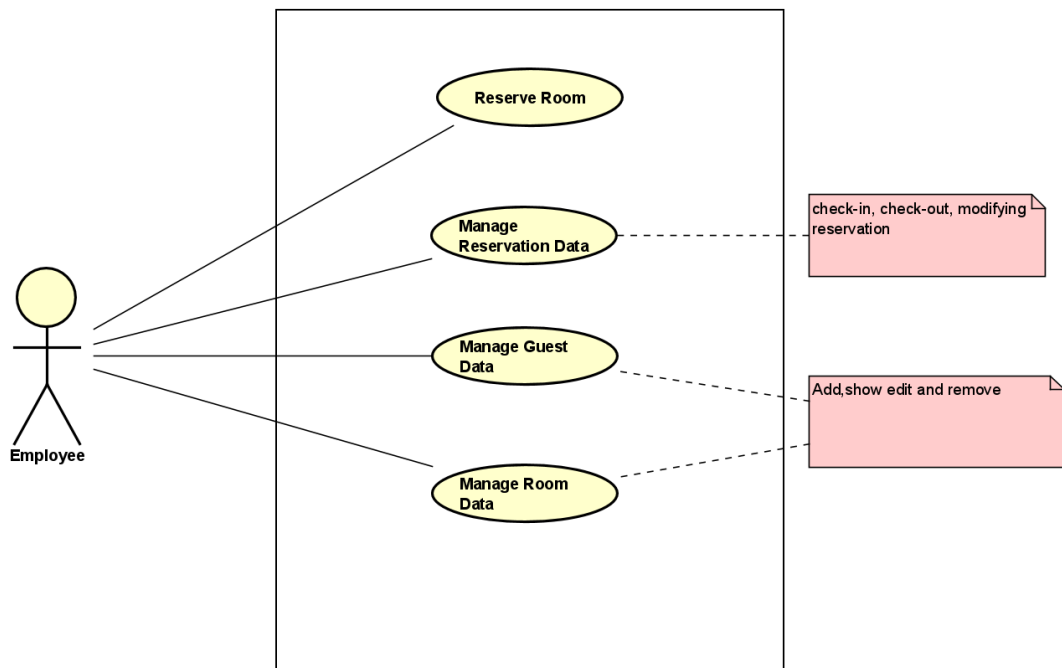9. As an employee, I want to add an additional bed, if the guest has requested it.

Low Priority:
10. As an employee, I want to add, modify, display or delete data about rooms,so rooms can be created, the information about the rooms can be modified or deleted and room information can be displayed.
11. As an employee, I want to update the rent price for the rooms, so that the prices of the rooms can be corrected.
12. As an employee, I want to set a room to unavailable if the room is undergoing refurbishment.
13. As an employee, I want to specify if smoking is permitted in the room, so the guest will be able to smoke in the room.
14. As an employee, I want to add an extra smoking fee to customers.
15. As an employee, I want to search for rooms that are available for smoking.
16. As an employee, I want to search for guests that are allowed to smoke.
17. As an employee, I want to search for frequent guests.

Non-functional requirements:

1. The program will be made in Java language.
2. The program can be interacted and input can be added by mouse or keyboard
3. The program will be easy and simple to use for the employees
4. The program is available for the employees when the reception has opening-hours.

Use Case diagram:

Use case description

Use case	Room Reservation
Summary	Create a reservation for a registered guest
Actor	Employee
Precondition	
Postcondition	A reservation with filled fields
Base sequence	1. Enter room type, check-in and check-out date 2. If dates are not today or future dates then go to step 1 again 3. System shows all available rooms for the period 4. Select room from the list 5. Add extras if guest demands 6. Adding guest to the reservation 7. Approve the reservation
Exception sequence	If there are no available rooms for the chosen period: Step 1-3 as base sequence

	If guest does not exist in the system database: create a new guest If guest data is outdated: request data from the guest
note	The reservation process can be cancelled at any time

Use case	Manage Reservation Data
Summary	Manage the requests of the guests. change of rooms. Keeping track of the number of guests
Actor	**Employee**
precondition	The reservation data is not up to date.
postcondition	The reservation data is updated
Base sequence	1. Enter the full name, phone number or email of the guest 2. The program shows the status of the booking for the guest, 3. The request of the guest such as new bed sheets, or a request for deliverables such as food or beverages will be registered in the system. 4. The request for the change of rooms will be done by the receptionist. 5. The staff will be notified about the guest request.
Exception sequence	
note	

Use case	Manage Guests
Summary	**Add,modify or delete data regarding guests**
Actor	**Employee**
precondition	Guest data is not up to date.
postcondition	**A customer has been added and is being assigned a room or removed from list of customers** Guest information has been updated, guest has been removed or added, onto said booking.
Base sequence	ADD: Adding new customer with values: 1a)Name(String) 1b)Phone(String) 1c)Address(String) 1d)Check-in-Date(String) 1e)Check-out-Date(String) System validates data entered and if wrong asks user to re-input data DISPLAY: 2a)Show a list of all customers 2b)Show a list of customers between two dates EDIT: Edit values of customers information DELETE: Delete a customer after verifying action 1. Search Booking 2. Check guest's name, phone number, address 3. Booker provides new Information. 4. Edit the guest with the new details provided by the guest. 5. Update the guest information and save.
Exception sequence	REMOVE GUEST Run Base Case(1-2) 3.Remove guest(s), provided by the booker from the reservation. 4.Update the guest in the booking and save. ADD GUEST Run Base Case(1-3)

	4. Create a new guest 5. Add the guest in the booking and save.
note	

Use case	Manage Room Data
Summary	Show room information and status or modify price per night
Actor	Employee
precondition	
postcondition	Data for a current room is modified
Base sequence	ATTRIBUTES: a)RoomNo(int) b)NoOfBeds(int) c)SizeOfBed(String) d)FloorNo(int) e)NoOfRooms(int) DISPLAY: Shows list of all occupied or unoccupied rooms and their information EDIT: Modify the values of each attribute of a room
Exception sequence	
note	