**Case Study: Hotel Management System**

**Context:**

You are required to develop a console-based Hotel Management System using Core Java. The application will allow hotel staff to manage rooms, reservations, and guest services. The application will utilize Java Collections for storing room and reservation information, implement Object-Oriented Programming principles, and include exception handling to manage potential errors.

**Requirements:**

**Features:**

1. **Add and Remove Rooms:**
   * Allow hotel staff to add new rooms to the system.
   * Allow hotel staff to remove rooms from the system.
2. **Update Room Details:**
   * Allow hotel staff to update details of existing rooms, such as room type, price, and availability.
3. **Check Room Availability:**
   * Allow hotel staff to check the availability of rooms for a given date range.
4. **Make Reservations:**
   * Allow hotel staff to make reservations for guests.
   * Ensure that reservations do not overlap with existing reservations for the same room.
5. **Check-in and Check-out Guests:**
   * Allow hotel staff to check-in guests to their reserved rooms.
   * Allow hotel staff to check-out guests from their rooms.

**Classes and Objects:**

1. **Room Class:**
   * Attributes: id, roomNumber, roomType, pricePerNight, isAvailable.
   * Methods: Constructors, getters and setters, toString method.
2. **Reservation Class:**
   * Attributes: id, roomId, guestName, startDate, endDate.
   * Methods: Constructors, getters and setters, toString method.
3. **HotelManagementSystem Class:**
   * Attributes: HashMap<Integer, Room> rooms, ArrayList<Reservation> reservations.
   * Methods:
     + addRoom(Room room): Adds a new room to the system.
     + removeRoom(int roomId): Removes a room from the system.
     + updateRoom(int roomId, Room updatedRoom): Updates details of an existing room.
     + checkRoomAvailability(int roomNumber, LocalDate startDate, LocalDate endDate): Checks the availability of a room for a given date range.
     + makeReservation(int roomId, String guestName, LocalDate startDate, LocalDate endDate): Makes a reservation for a guest.
     + checkInGuest(int roomId, String guestName): Checks-in a guest to a reserved room.
     + checkOutGuest(int roomId): Checks-out a guest from a room.
     + Helper methods for input validation and exception handling.

**Deliverables:**

1. Complete source code for the Hotel Management System.
2. Documentation including:
   * How to run the application.
   * Instructions for each feature.
   * Explanation of the exception handling implemented.
3. A brief report on the application design and how Object-Oriented principles were applied.