Hotel Booking & Reservation System

1️⃣Users

Roles: GUEST and ADMIN

Purpose: - Guests can browse hotels, book rooms, view/cancel their bookings. - Admins can add hotels, add rooms, and manage all bookings and users.

Fields: - id → Unique identifier - name → Guest or admin name - email → Unique login/email - role → Guest or Admin

Relations: - bookings → One-to-many relation to bookings table

2️⃣Hotels

Purpose: Store hotel information and rooms

Fields: - id → Unique identifier - name → Hotel name - location → City or area - description → Optional

Relations: - rooms → One-to-many relation to Room table

3️⃣Rooms

Purpose: Represent individual rooms in hotels

Fields: - id → Unique identifier - number → Room number - type → SINGLE, DOUBLE, SUITE - pricePerNight → Room cost - isAvailable → Tracks availability

Relations: - hotelId → Room belongs to a hotel - bookings → One-to-many relation to bookings

4️⃣Bookings

Purpose: Track which user booked which room for which dates

Fields: - id → Unique identifier - checkInDate / checkOutDate → Booking duration - status → PENDING, CONFIRMED, CANCELLED

Relations: - userId → The guest who booked the room - roomId → The room that was booked

5️⃣API Flow

Users: - POST /api/users → Create user - GET /api/users/:id → Fetch user info - PUT /api/users/:id → Update user info - DELETE /api/users/:id → Remove user

Hotels: - POST /api/hotels → Add new hotel (admin only) - GET /api/hotels → Fetch all hotels - GET /api/hotels/:id → Fetch single hotel details - PUT /api/hotels/:id → Update hotel info (admin only) - DELETE /api/hotels/:id → Delete hotel (admin only)

Rooms: - POST /api/rooms → Add room to hotel (admin only) - GET /api/rooms → Fetch all rooms (with optional filters: hotel, type, price) - GET /api/rooms/:id → Fetch room details - PUT /api/rooms/:id → Update room info (admin only) - DELETE /api/rooms/:id → Delete room (admin only)

Bookings: - POST /api/bookings → Book a room (guest) - GET /api/bookings → Fetch all bookings (admin sees all, guest sees own) - GET /api/bookings/:id → Fetch single booking - PUT /api/bookings/:id → Update booking (e.g., change dates or status) - DELETE /api/bookings/:id → Cancel booking

6️⃣Key Features / Logic

- Availability Check: Only show rooms that are available for the requested dates.
- Booking Conflict: Prevent double booking the same room.
- Role-based access: Guests cannot modify hotels or rooms.
- Price filtering: Guests can search rooms within a price range.
- Admin Dashboard: For adding hotels, rooms, and managing bookings.

7️⃣Possible Extensions (Optional)

- Payment Integration: Add Stripe/PayPal for bookings
- Search & Filters: Location, price, room type
- Ratings & Reviews: Guests can leave reviews for hotels
- Notifications: Email alerts on booking confirmation or cancellation

8️⃣Additional Recommendations (added)

- Use pagination for large lists of hotels, rooms, and bookings.
- Implement input validation for dates and numeric fields.
- Consider adding a basic frontend dashboard with Tailwind CSS or React.
- Add logging for API requests and errors for debugging.