

User Model

- The User model represents all users in the system, including visitors, hotel owners, and general users.
- One-to-Many relationship with Hotel, Booking, Itinerary, Payment and Notification model, this means that a user can own multiple hotels, bookings, itineraries, payments, and notifications.
- One-to-One relationship with UserPreference, this means that a user can only have one preferences record.

UserPreferences Model

- The UserPreferences Model stores user-specific settings
- One-to-One relationship linked to User

Hotel Model

- Hotel model Represents hotels in the system.
- One-to-Many relationship with Room, Booking and HotelImages, represent that one hotel can have multiple rooms, booking and the hotel image.

Hotel Image Model

- The hotel image model stores images of hotels.
- Many-to-One relationship with a Hotel, that means each image is associated with a single hotel.

Room Model

- Room model represents different room types in a hotel
- One-to-Many relationship with RoomImages, Booking and RoomAvailability model representing that A room can have multiple images, bookings, and availability records.
- Many-to-One relationship with a Hotel, representing that one room belongs to one Hotel.

RoomAvailability Model

- Tracks room availability on specific dates.
- Many-to-One with Room which means each availability record is linked to one Room.

RoomImage Model

- Representing image for each room
- Many-to-One relationship with Room which means that each room image belongs to only one room.

Booking Model

- The Booking model stores all reservations
- Many-to-One relationship with User, Hotel and Room, representing that Each booking is associated with one user, hotel and room.
- Many-to-One relationship with Itinerary, which means that each booking can be a part of a Itinerary.

Itinerary Model

- The Itinerary model represents a collection of bookings grouped into a single trip.
- One-to-Many relationship with Booking, Payment and Invoice which means that an itinerary can have multiple bookings, payments, and invoices.

- Many-to-One relationship with User, which means that an itinerary belongs to one user.

Payment Model

- Payment model stores payment transactions
- Many-to-One relationship with User and Itinerary, which means that each payment is linked to one user and one itinerary.

Invoice Model

- Invoice model stores invoice records for bookings and payments.
- Many-to-One relationship with User and Itinerary which means that each invoice is linked to one user and one itinerary.

Notification Model

- Notification model stores system notifications for users.
- Many-to-One relationship with User, which means that each notification is linked to a single user.

ER diagram:

- Diagram can also be find under my-next-backend/prisma/ERD.svg

