

NexaRide - Bus Booking System: ER Diagram & Database Schema

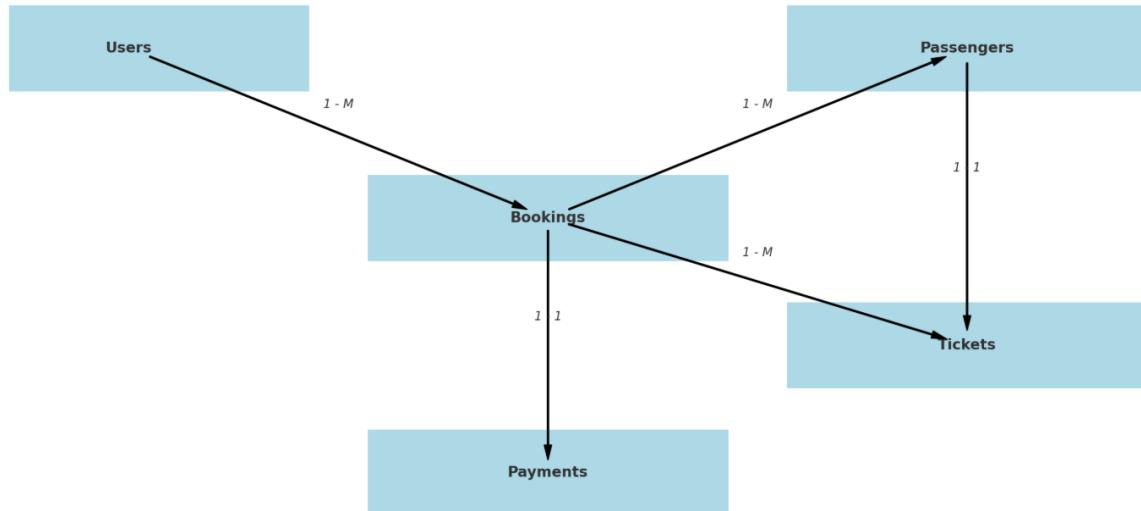
Project Overview

NexaRide is a Java-based bus booking service that handles user registrations, various types of bookings (one-way, two-way, tourism), passenger details, payments, and ticket generation.

This document explains the **Entity Relationship Diagram (ERD)** and provides the complete **database schema** needed to support the backend operations of the NexaRide application.

ER Diagram

ER Diagram stands for **Entity-Relationship Diagram**. It is a visual representation used in database design to show how data is structured and how different pieces of data relate to each other.



Entities and Their Relationships:

1. **Users**

- `user_id` (PK)

- name, address, email, password, mobile_no

2. Bookings

- booking_id (PK)
- FK to users.user_id
- booking_type, bus_type, from_location, to_location, date_of_journey, return_from, return_to, return_date, tourism_place, number_of_days, number_of_persons, total_price

3. Passengers

- passenger_id (PK)
- FK to bookings.booking_id
- name, age, address, mobile_no, email_id, aadhaar_no

4. Payments

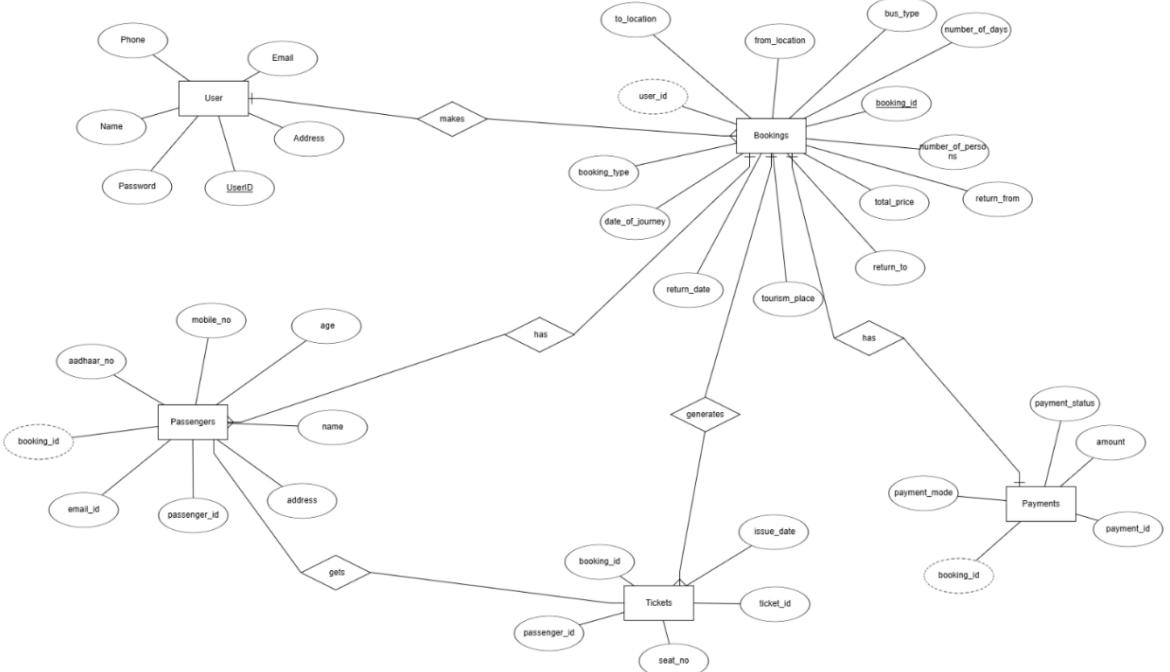
- payment_id (PK)
- FK to bookings.booking_id
- payment_mode, amount, payment_status

5. Tickets

- ticket_id (PK)
- FK to passengers.passenger_id
- FK to bookings.booking_id
- seat_no, issue_date

Relationships:

- **Users → Bookings:** 1-to-Many
- **Bookings → Passengers:** 1-to-Many
- **Bookings → Payments:** 1-to-1
- **Bookings → Tickets:** 1-to-Many
- **Passengers → Tickets:** 1-to-1



Passenger Data :

```

SELECT t.ticket_id, p.name AS passenger_name, b.from_location, b.to_location, t.seat_no,
t.issue_date
FROM tickets t
JOIN passengers p ON t.passenger_id = p.passenger_id
JOIN bookings b ON t.booking_id = b.booking_id;
    
```

	ticket_id	passenger_name	from_location	to_location	seat_no	issue_date
▶	1	Rahul	HYD	Vijayawada	A1	2025-08-02 10:38:59
	2	Arun	AP	Vizag	B1	2025-08-02 10:38:59
	3	Kiran	AP	Vizag	B2	2025-08-02 10:38:59
	4	Varun	NULL	NULL	C1	2025-08-02 10:38:59
	5	Prasad	MUMBAI	Pune	D1	2025-08-02 10:38:59

Payment Data :

```

SELECT pay.payment_id, u.name AS user_name, b.booking_id, pay.payment_mode,
pay.amount, pay.payment_status
FROM payments pay
JOIN bookings b ON pay.booking_id = b.booking_id
JOIN users u ON b.user_id = u.user_id;
    
```

	payment_id	user_name	booking_id	payment_mode	amount	payment_status
▶	1	Rahul Sharma	1	upi	1200	done
	2	Arun Kumar	2	netbanking	1800	done
	3	Kiran Reddy	3	debit card	10000	done
	4	Varun Singh	4	credit card	800	done
	5	Prasad Joshi	5	upi	3000	done

Booking Data :

```
SELECT p.passenger_id, p.name AS passenger_name, b.booking_id, b.from_location,
b.to_location, b.date_of_journey
FROM passengers p
JOIN bookings b ON p.booking_id = b.booking_id;
```

	passenger_id	passenger_name	booking_id	from_location	to_location	date_of_journey
▶	1	Rahul	1	HYD	Vijayawada	2024-08-10
	2	Arun	2	AP	Vizag	2024-08-15
	3	Kiran	2	AP	Vizag	2024-08-16
	4	Varun	3	NULL	NULL	NULL
	5	Prasad	4	MUMBAI	Pune	2024-08-12

Ticket Data :

```
SELECT b.booking_id, u.name AS user_name, b.booking_type, b.bus_type,
b.from_location, b.to_location, b.date_of_journey, b.total_price
FROM bookings b
JOIN users u ON b.user_id = u.user_id;
```

	booking_id	user_name	booking_type	bus_type	from_location	to_location	date_of_journey	total_price
▶	1	Rahul Sharma	one way	ac	HYD	Vijayawada	2024-08-10	1200
	2	Arun Kumar	two way	non-ac	AP	Vizag	2024-08-16	1800
	3	Kiran Reddy	tourism	ac-sleeper	NULL	NULL	NULL	10000
	4	Varun Singh	one way	ac	MUMBAI	Pune	2024-08-12	800
	5	Prasad Joshi	two way	ac-sleeper	Bangalore	Mysore	2024-08-14	3000

Conclusion

In this project, we designed the data model for NexaRide by identifying the key entities: **User, Passengers, Booking, Payment, and Tickets.** We defined each entity's attributes and mapped out the relationships among them. Using this information, we created an ER diagram that accurately represents how these entities interact within the system. This diagram provides a clear structure for building the database to support NexaRide's core services such as user management, passenger details, booking processes, payment handling, and ticket issuance.