Bus Reservation System

By

Sai Charitha Pathipati

# Introduction

The Bus Reservation System streamlines bus ticket booking, offering passengers an intuitive platform to search routes, check seat availability, and book tickets online. It ensures real-time updates on seat availability, enabling informed decisions. Secure payment gateways facilitate transactions, accommodating various payment methods. Administrators manage routes, schedules, and passenger data efficiently. Passengers can cancel tickets with automated refund processing. The system generates reports for route performance and revenue analysis, aiding strategic decision-making. In essence, it enhances the booking experience, optimizes operations, and boosts customer satisfaction in the transportation sector.

# **Requirements**

## Operations to be captured:

* **Passenger Management:** The system facilitates the addition of new passenger information. Passenger details are editable, allowing for updates as needed. Users can search for passengers using criteria like name or ID.
* **Ticket Management:** Users are able to issue new tickets, complete with reservation features. Cancellation of tickets is supported within the system. Ticket details are accessible for review.
* **Payment Processing:** The system processes payments for tickets, accepting methods like credit card or debit card.
* **Bus Management:** New buses can be added to the system as needed. Bus information can be modified and updated. Users can view comprehensive bus details such as total seats and availability.
* **Route** **Management:** Administrators can create new routes within the system. Routes are editable, enabling modifications to stop or other details. Detailed route information, including schedules and stops, is available for viewing.

## Information to be tracked.

* **Passenger Information:** PassengerID (unique identifier), Name, Age, Gender, ContactNumber, Address.
* **Ticket** **Information:** TicketNumber (unique identifier), Price, Date of Booking.
* **Payment Information**: PaymentID (unique identifier), PaymentType (e.g., credit card, debit card), CardNumber.
* **Bus** **Information:** BusNumber (unique identifier), TotalSeats, AvailableSeats, SourceStop, DestinationStop, ArrivalTime, Destination.
* **Route Information:** RouteID (unique identifier), PickupPoints.

## Relationships and Constraints

* **Passenger – Ticket Relationships:** One passenger can buy many tickets (one-to-many relationship). A ticket belongs to one passenger (many-to-one relationship).
* **Ticket – Payment Relationships**: One ticket has one payment (one-to-one relationship). A payment belongs to one ticket (one-to-one relationship).
* **Bus – Route Relationships:** Bus and Route: This relationship can be interpreted in two ways depending on the business scenario.
* Many buses travel through many routes (many-to-many relationship). This might be the case if a bus can be assigned to different routes at various times. One bus travel through one route at a time (one-to-many relationship). This might be the case if a route always has multiple buses assigned to it, such as during rush hour**.**
* **Route – Ticket Relationships:** One route can have many tickets (one-to-many relationship). A ticket belongs to one route (many-to-one relationship).
* **Constraints:** Passenger ID in Ticket Information table references Passenger ID in Passenger Information table. Bus Number in Ticket Information table references Bus Number in Bus Information table. Passenger ID, Ticket Number, Payment ID, and Route ID must be unique identifiers.

# ER Diagram:

A diagram of a flowchart

Description automatically generated

## Entities

1. **Passenger Entity:** Represents individuals with attributes including Passenger Id (Primary Key), Name, Age, Gender, Address and ContactNumber.
2. **Ticket Entity:** Represents booked tickets with attributes including TicketNo (Primary Key), Price, and Ticket Date.
3. **Payment Entity:** Represents payments with attributes including PaymentID (Primary Key), PaymentType, CardNo.
4. **Bus Entity:** Represents buses with attributes including BusID (Primary Key), Total Seats, AvailableSeats, SourceStop, DestinationStop, ArrivalTime, Destination.
5. **Route Entity:** Represents Routes with attributes includingRouteID (unique identifier), PickupPoints.

## Relationships and Cardinalities:

**Passenger and Ticket:** A passenger can purchase one or multiple tickets (One-to-Many). Each ticket is associated with one passenger (Many-to-One).

**Ticket and Payment:** Each ticket is linked to one payment (One-to-One). Each payment corresponds to one ticket (One-to-One).

**Route and Bus:** A route can accommodate multiple buses (One-to-Many). A bus can operate on various routes (Many-to-Many).

**Constraints**: PassengerID, TicketNo, PaymentID, and RouteID must serve as unique identifiers (Primary Key).

# Relational Schema

A diagram of a bus terminal

Description automatically generated

The relational schema for the bus reservation System Database consists of five tables: passenger, tickets, Payment, bus and route.

* The **Passenger** table stores information about each Passenger, including a unique ID, name, contact information, and address.
* The **Ticket** table records details about each ticket, such as the TicketNo, price, ticket date and payment status. It’s linked to the passenger table through the TicketNo, which is a foreign key in the ticket table.
* The **Payment** table tracks payments made by passenger. It includes details like the payment Type, Card No. This table is connected to the Ticket table through the PaymentID, a foreign key in the ticket table.
* The **Route** table keeps track of the bus route for each passenger. It’s linked to the bus table through the RouteID.
* The **Bus** table gives the information about the bus no, seat no, total seats, source, destination. It’s linked to the passenger and ticket tables through the bus no.

# Referential integrity constraints

**Update Constraints:**

* If a PassengerID in the passenger table is updated, it should also be updated in the ticket table to maintain consistency.
* If a bus no in the Bus table is updated, it should also be updated in the ticket and passenger table.

**Delete Constraints:**

* If a bus no record is deleted from the bus table, all associated records in the ticket and passenger tables should also be deleted to avoid orphan records.
* If a RouteID record is deleted from the route table, all associated records in the bus table should also be deleted.
* If a PassengerID record is deleted from the passenger table, all associated records in the ticket table should also be deleted.

**User-Defined Constraints:**

* Age must be greater than zero.
* Passengers cannot book tickets for buses that are already at full capacity.
* Ensuring that a ticket can only be associated with a bus that is currently active.
* The price in the ticket table should always be greater than zero.

# Implementation SQL

CREATE TABLE Route

(

RouteID INT NOT NULL,

PickupPoint INT NOT NULL,

PRIMARY KEY (RouteID)

);

CREATE TABLE Payment

(

PaymentID INT NOT NULL,

PaymentType INT NOT NULL,

CardNo INT NOT NULL,

PRIMARY KEY (PaymentID)

);

CREATE TABLE Bus

(

TotalSeats INT NOT NULL,

Bus.No INT NOT NULL,

AvailableSeats INT NOT NULL,

SeatNo INT NOT NULL,

ArrivalTime INT NOT NULL,

Source INT NOT NULL,

Destination INT NOT NULL,

RouteID INT NOT NULL,

PRIMARY KEY (Bus.No),

FOREIGN KEY (RouteID) REFERENCES Route(RouteID)

);

CREATE TABLE Passenger

(

PassengerID INT NOT NULL,

Gender INT NOT NULL,

Age INT NOT NULL,

Name INT NOT NULL,

Address INT NOT NULL,

ContactNumber INT NOT NULL,

Bus.No INT NOT NULL,

PRIMARY KEY (PassengerID),

FOREIGN KEY (Bus.No) REFERENCES Bus(Bus.No)

);

CREATE TABLE Tickets

(

TicketNo INT NOT NULL,

Price INT NOT NULL,

TicketDate INT NOT NULL,

PassengerID INT NOT NULL,

Bus.No INT NOT NULL,

PaymentID INT NOT NULL,

PRIMARY KEY (TicketNo)

FOREIGN KEY (PassengerID) REFERENCES Passenger (PassengerID),

FOREIGN KEY (Bus.No) REFERENCES Bus (Bus.No),

FOREIGN KEY (PaymentID) REFERENCES Payment (PaymentID)

);

-- Insert 10 records into the Route table

INSERT INTO Route (RouteID, PickupPoint)

VALUES

(1, 101),

(2, 102),

(3, 103),

(4, 104),

(5, 105),

(6, 106),

(7, 107),

(8, 108),

(9, 109),

(10, 110);

-- Insert 10 records into the Payment table

INSERT INTO Payment (PaymentID, PaymentType, CardNo)

VALUES

(1, 201, 1234567812345678),

(2, 202, 9876543210987654),

(3, 203,135689090900),

(4, 204, 'paypal@example.com'),

(5, 205, 'googlepay@example.com'),

(6, 206, 'applepay@example.com'),

(7, 207, 987654321),

(8, 208, 'venmo@example.com'),

(9, 209, 'bitcoin:1BitcoinAddress'),

(10, 210, 123456789);

-- Insert 10 records into the Bus table

INSERT INTO Bus (TotalSeats, BusNo, AvailableSeats, SeatNo, ArrivalTime, Source, Destination, RouteID)

VALUES

(50, 1, 50, 1, 800, 101, 201, 1),

(55, 2, 55, 2, 900, 102, 202, 2),

(60, 3, 60, 3, 1000, 103, 203, 3),

(65, 4, 65, 4, 1100, 104, 204, 4),

(70, 5, 70, 5, 1200, 105, 205, 5),

(75, 6, 75, 6, 1300, 106, 206, 6),

(80, 7, 80, 7, 1400, 107, 207, 7),

(85, 8, 85, 8, 1500, 108, 208, 8),

(90, 9, 90, 9, 1600, 109, 209, 9),

(95, 10, 95, 10, 1700, 110, 210, 10);

-- Insert 10 records into the Passenger table

INSERT INTO Passenger (PassengerID, Gender, Age, Name, Address, ContactNumber, BusNo)

VALUES

(1, 1, 30, 101, 201, 1234567890, 1),

(2, 2, 25, 102, 202, 9876543210, 2),

(3, 1, 40, 103, 203, 5555555555, 3),

(4, 2, 35, 104, 204, 4444444444, 4),

(5, 1, 28, 105, 205, 7777777777, 5),

(6, 2, 45, 106, 206, 8888888888, 6),

(7, 1, 32, 107, 207, 6666666666, 7),

(8, 2, 27, 108, 208, 3333333333, 8),

(9, 1, 22, 109, 209, 2222222222, 9),

(10, 2, 38, 110, 210, 9999999999, 10);

-- Insert 10 records into the Tickets table

INSERT INTO Tickets (TicketNo, Price, TicketDate, PassengerID, BusNo, PaymentID)

VALUES

(101, 25, '2024-05-01', 1, 1, 1),

(102, 30, '2024-05-02', 2, 2, 2),

(103, 35, '2024-05-03', 3, 3, 3),

(104, 40, '2024-05-04', 4, 4, 4),

(105, 45, '2024-05-05', 5, 5, 5),

(106, 50, '2024-05-06', 6, 6, 6),

(107, 25, '2024-05-01', 7, 7, 7),

(108, 30, '2024-05-02', 8, 8, 8),

(109, 35, '2024-05-03', 9, 9, 9),

(110, 40, '2024-05-04', 10, 10, 10);

# Datawarehouse Requirements

Subject of Analysis: The subject of analysis, the bus reservation system, encompasses various facets vital for operational efficiency and customer satisfaction. Analysing passenger demographics, ticket sales patterns, bus occupancy rates, and payment methods enables insights into travel behaviour, route popularity, and revenue generation, optimizing service provision and decision-making processes.

**Useful Fields or Attributes:** The following fields from our database will be useful in the analysis:

**Passenger**: PassengerID, Name, Age, Gender, ContactNumber, Address.

**Ticket**: TicketNo, PassengerID, Price, Ticket Date.

**Payment**: PaymentID, PaymentType, CardNo

**Bus**: TotalSeats, BusNo, AvailableSeats, SeatNo, ArrivalTime, Source, Destination.

**Route**: RouteID, PickupPoint.

**Granularity of the Facts/Measures:** The granularity of facts/measures in the bus reservation system analysis varies, encompassing individual passenger details, ticket sales data, bus occupancy statistics, route utilization patterns, and payment transaction records. This allows for detailed insights into passenger behaviour, operational efficiency, and revenue generation, facilitating informed decision-making processes.

# Star Schema of Data Warehouse

A diagram of a server

Description automatically generated

# Conclusion:

This study of the bus ticketing system outlined essential needs for both the operational database and a prospective data warehouse. The relational schema ensures data integrity and streamlines management of passengers, tickets, buses, and route. The implemented SQL statements establish necessary tables with suitable data types and constraints. Additionally, the analysis of data warehouse requirements highlighted the potential to examine passenger booking trends. The proposed star schema, incorporating dimensions such as Passenger (demographics), Date (time-based), and Bus Route (destination), alongside the Ticket Sales fact table with revenue and ticket count measures, offers insights for strategic decisions on pricing, resource allocation, and route optimization.