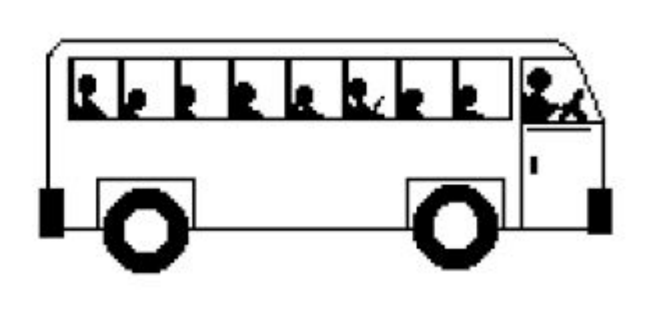
Bus Booking Management System

●●● DBMS Course Project



Team Members

Balne Niteesha	18114017
Diksha	18114019
Karan Singh	18114035
Kavya Barnwal	18114039
Khushi	18114040

Introduction

Traveling is a large growing business across all countries. We observed the working of the Bus reservation system and after going through it, we got to know that there are many operations, which they have to do manually. It takes a lot of time and causes many errors while data entry. Due to this, sometimes a lot of problems occur and they are facing many disputes with customers. To solve the above problem, and further maintain records of passenger details, seat availability, bus availability and other things, we are offering this proposal of a computerized reservation system.

Project Description

The focus of the project is to computerize traveling companies to manage data, details of customers, details of various buses so as to ease companies' tasks and shifting the ticket booking process to an online platform for the easier access of customers. It reduces the possibility of errors or any discrepancy in any kind of data. It replaces all the paperwork.

This bus booking management system has three modules.:

- First module helps the customer to login or register into the system.
- Second module helps to inquire about the availability of seats in a particular bus at a particular date. It helps him to reserve a ticket or cancel a reserved ticket.
- Third module allows the admin of the system to add seats to various buses and to view customers databases.

Features of the system

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features:

✓Availability of seats can be enquired very easily.

✓Passengers can also cancel their tickets easily.

✓Minimum time needed for the various processing

✓Better Service

✓Ensures data accuracy.

✓Records are efficiently maintained by DBMS.

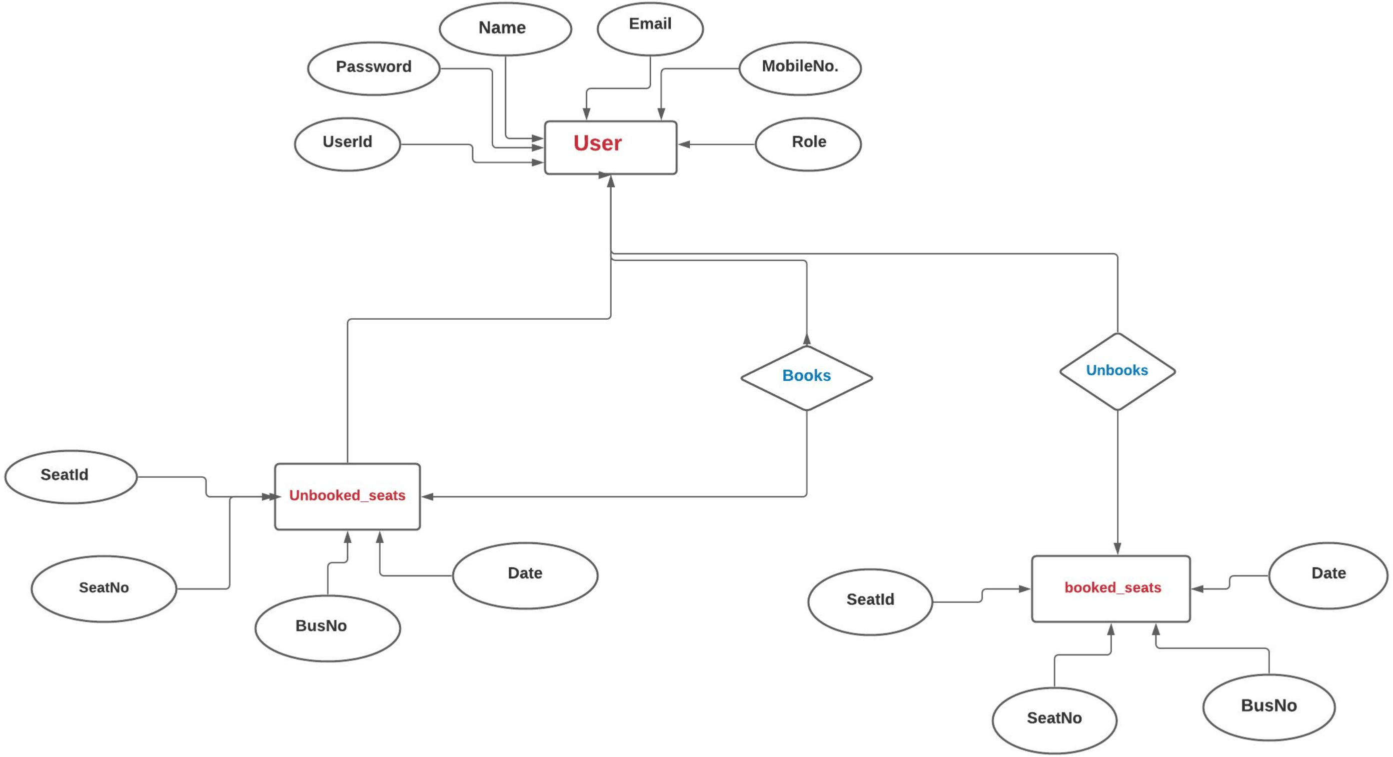
✓DBMS also provides security for the information.

✓Any person across the world, with required setup can access this service.

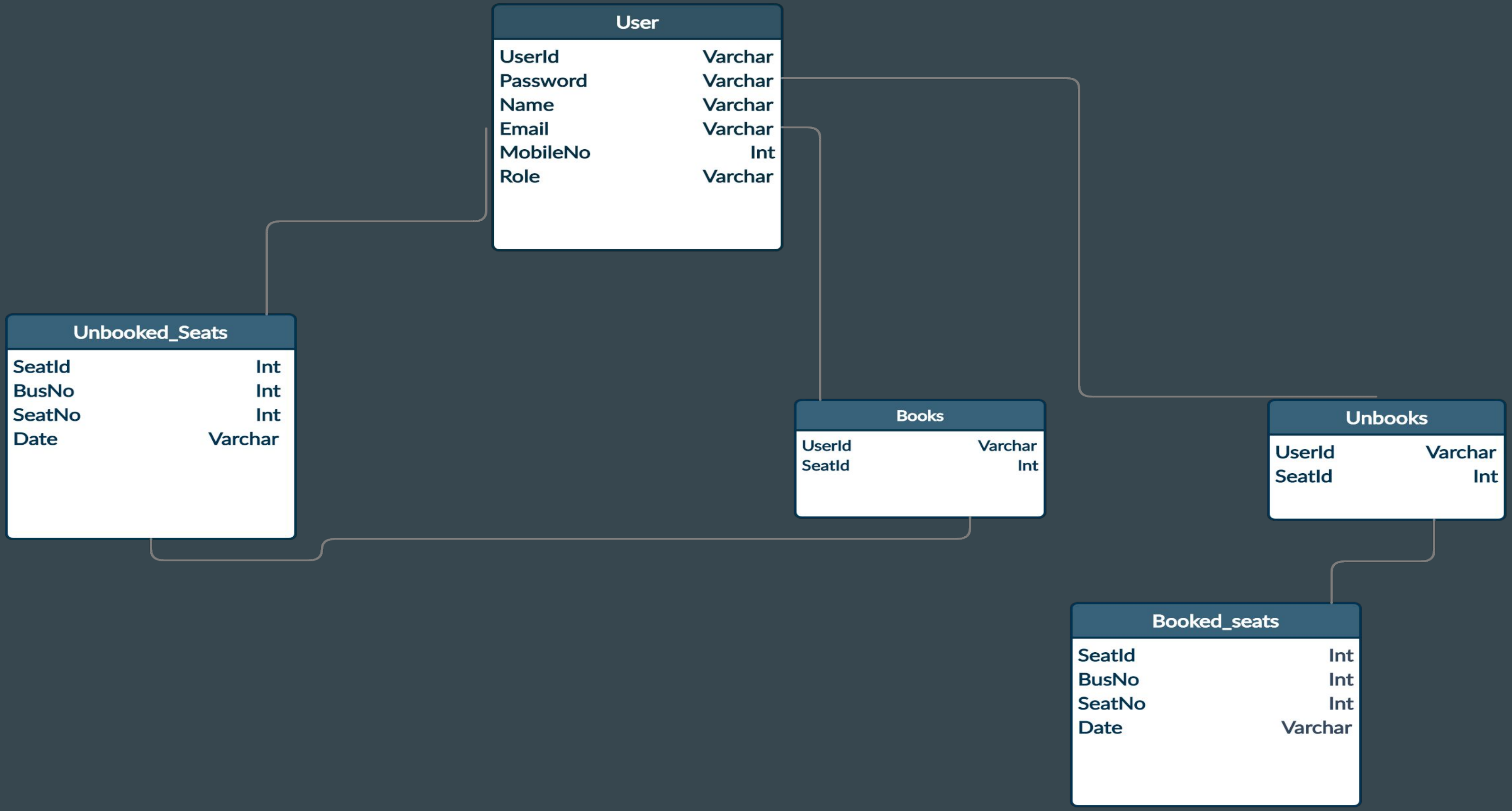
✓This would help the corporation prepare and organize its schedules more efficiently on the basis of traffic demand.

- **Software Requirements:**
  - Operating system : Windows (optional)
  - Java Virtual Machine
  - Front end : Java Runtime
  - Integrated Development Environment(IDE) : Netbeans
  - Server to host the database on machine : XAMPP
  - JDBC driver : MySQL Connector

ER Diagram


Name
Email
Password
MobileNo.
UserId
User
Role
Books
Unbooks
SeatId
Unbooked_seats
SeatNo
Date
BusNo
SeatId
booked_seats
Date
SeatNo
BusNo

Relational Schema:



User	
UserId	Varchar
Password	Varchar
Name	Varchar
Email	Varchar
MobileNo	Int
Role	Varchar

Unbooked_Seats	
SeatId	Int
BusNo	Int
SeatNo	Int
Date	Varchar

Books	
UserId	Varchar
SeatId	Int

Unbooks	
UserId	Varchar
SeatId	Int

Booked_seats	
SeatId	Int
BusNo	Int
SeatNo	Int
Date	Varchar

Queries

Select

σ

Insert

→

Update

Delete

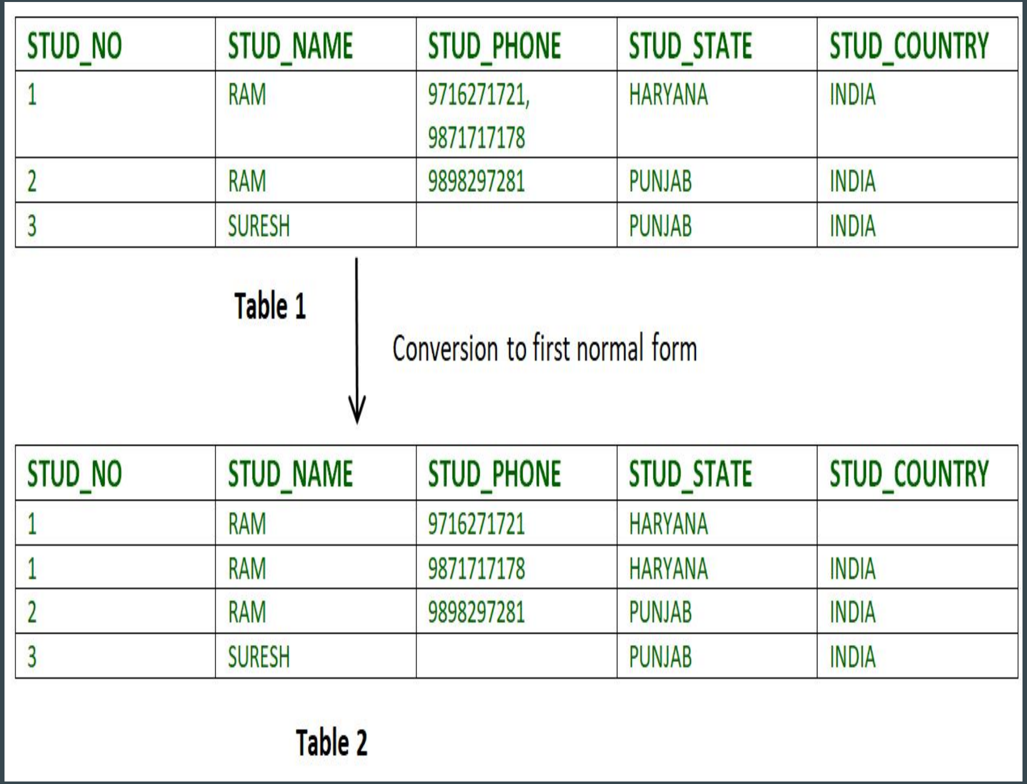
Database and Normalisations:

- Database Normalisation is the process of organizing attributes of the database to reduce or eliminate “data redundancy” .

- Redundancy in database may cause insertion , deletion and updation anomalies.

1NF :

- A relation is in first normal form if every attribute is a single valued attribute.



STUD_NO	STUD_NAME	STUD_PHONE	STUD_STATE	STUD_COUNTRY
1	RAM	9716271721, 9871717178	HARYANA	INDIA
2	RAM	9898297281	PUNJAB	INDIA
3	SURESH		PUNJAB	INDIA

Table 1

Conversion to first normal form

STUD_NO	STUD_NAME	STUD_PHONE	STUD_STATE	STUD_COUNTRY
1	RAM	9716271721	HARYANA	
1	RAM	9871717178	HARYANA	INDIA
2	RAM	9898297281	PUNJAB	INDIA
3	SURESH		PUNJAB	INDIA

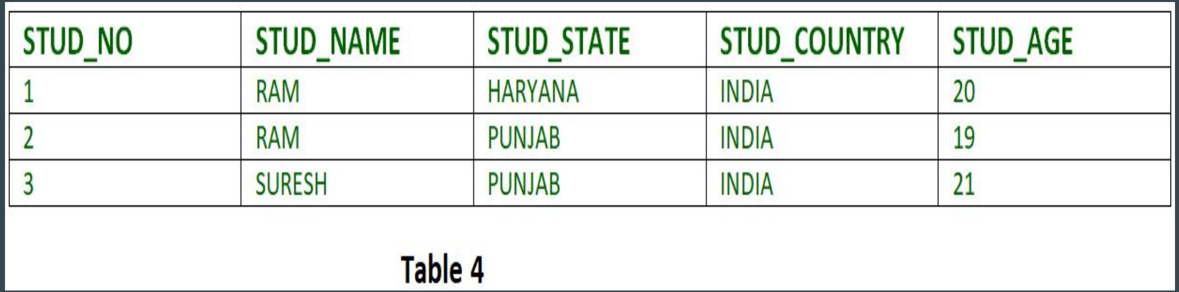
Table 2

2 NF:

- Relation should be in 1NF .
- No “non-prime” attribute should be dependent on a proper subset of candidate key.

3NF :

- Relation should be in 2NF .
- There should not be any “transitive dependency”.



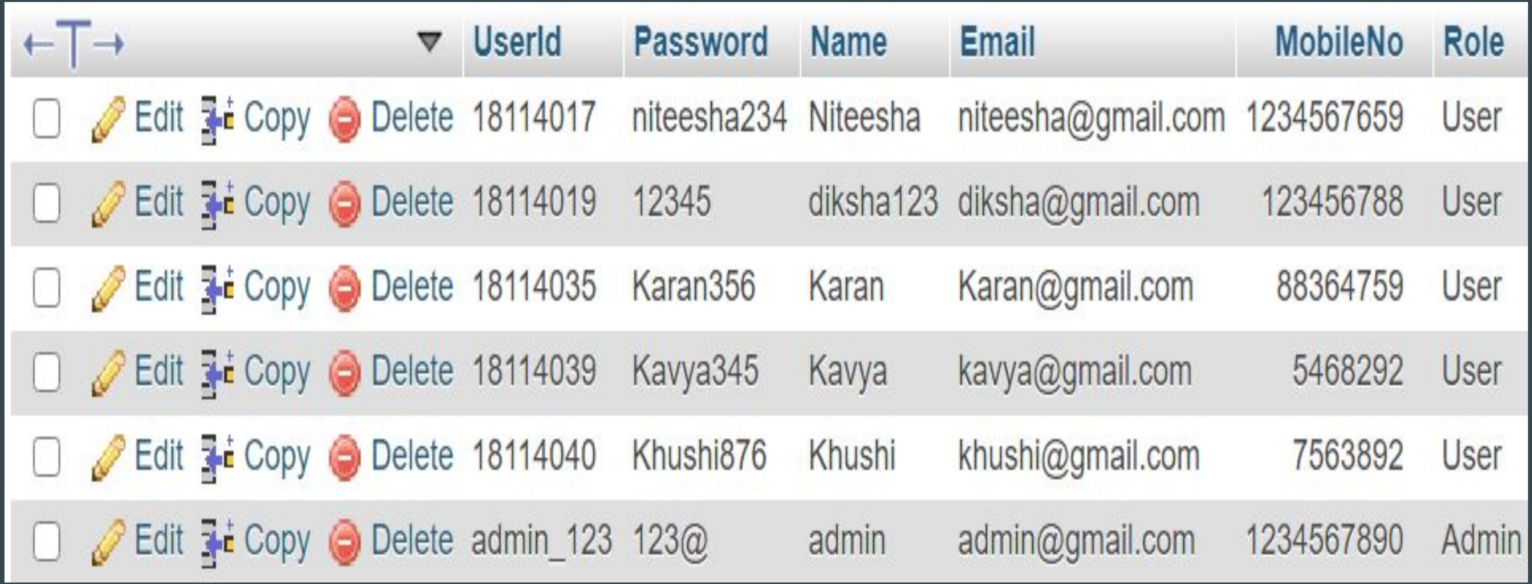
STUD_NO	STUD_NAME	STUD_STATE	STUD_COUNTRY	STUD_AGE
1	RAM	HARYANA	INDIA	20
2	RAM	PUNJAB	INDIA	19
3	SURESH	PUNJAB	INDIA	21

Table 4

BCNF:

- Relation should be in 3NF.
- For every non-trivial dependency , attribute should be dependent on super key.

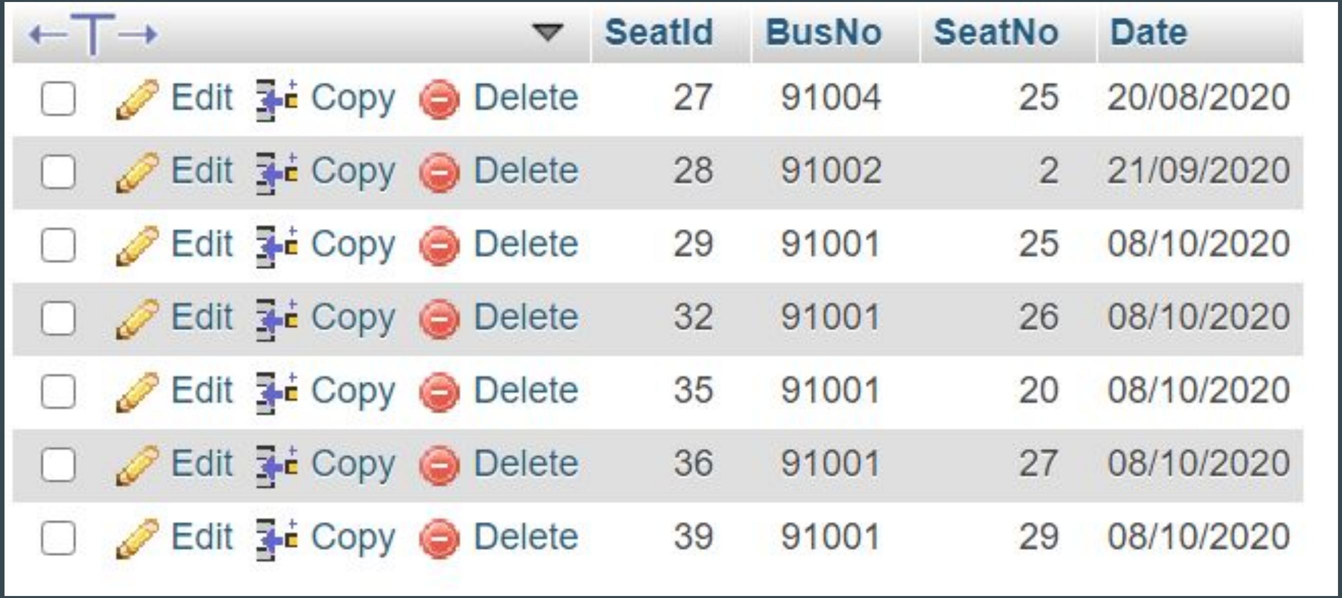
User Table :



←T→			▼	UserId	Password	Name	Email	MobileNo	Role
☐	Edit	Copy	Delete	18114017	niteesha234	Niteesha	niteesha@gmail.com	1234567659	User
☐	Edit	Copy	Delete	18114019	12345	diksha123	diksha@gmail.com	123456788	User
☐	Edit	Copy	Delete	18114035	Karan356	Karan	Karan@gmail.com	88364759	User
☐	Edit	Copy	Delete	18114039	Kavya345	Kavya	kavya@gmail.com	5468292	User
☐	Edit	Copy	Delete	18114040	Khushi876	Khushi	khushi@gmail.com	7563892	User
☐	Edit	Copy	Delete	admin_123	123@	admin	admin@gmail.com	1234567890	Admin

- Contains the information of customers and admin.
- No partial dependency , no transitive dependency and no multivalued dependency .
- Normalised upto BCNF Form .

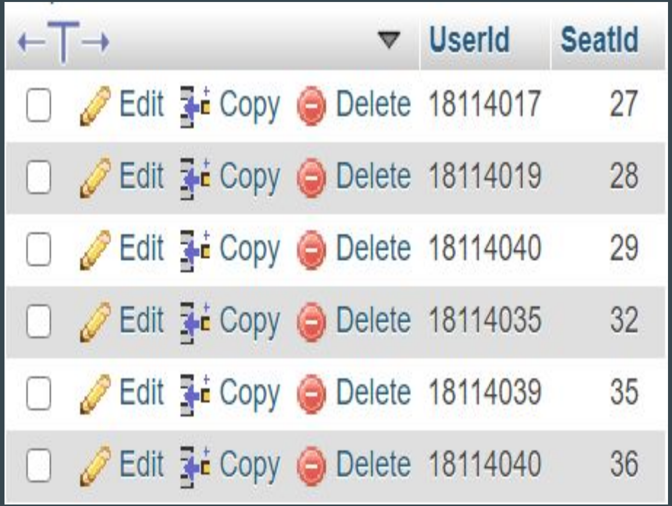
Booked_seats Table:



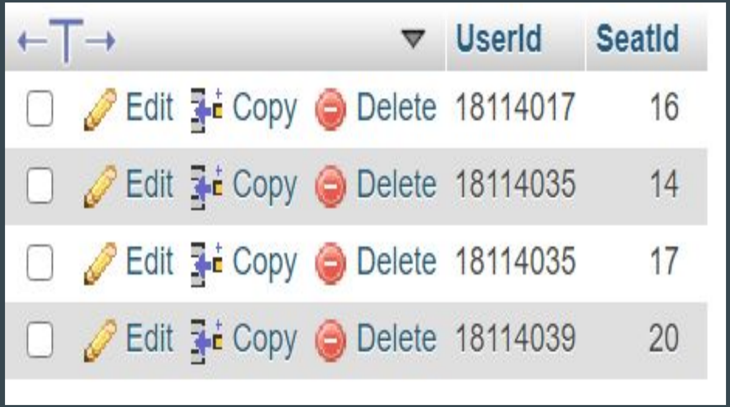
SeatId	BusNo	SeatNo	Date
27	91004	25	20/08/2020
28	91002	2	21/09/2020
29	91001	25	08/10/2020
32	91001	26	08/10/2020
35	91001	20	08/10/2020
36	91001	27	08/10/2020
39	91001	29	08/10/2020

- Contains the information about the seats which are already booked by some passenger.
- Normalised upto BCNF Form .

Books and Unbooks Table:



←T→				▼	UserId	SeatId
□	Edit	Copy	Delete		18114017	27
□	Edit	Copy	Delete		18114019	28
□	Edit	Copy	Delete		18114040	29
□	Edit	Copy	Delete		18114035	32
□	Edit	Copy	Delete		18114039	35
□	Edit	Copy	Delete		18114040	36



←T→				▼	UserId	SeatId
□	Edit	Copy	Delete		18114017	16
□	Edit	Copy	Delete		18114035	14
□	Edit	Copy	Delete		18114035	17
□	Edit	Copy	Delete		18114039	20

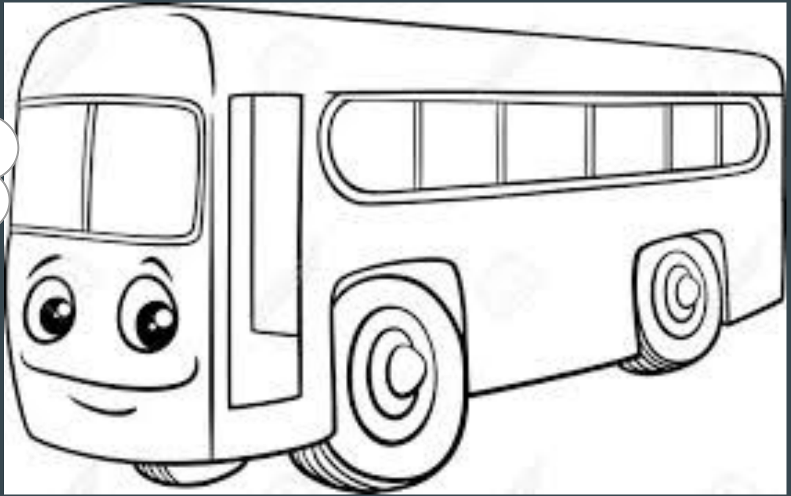
- The Books table contains information about which user booked which seat , and accordingly updates the booked_seats table.
- The Unbooks table contains information about which user unbooked which seat , and accordingly updates the unbooked_seats table.
- Both attributes are primary key , so no transitive , multivalued or partial dependencies.
- Normalised upto BCNF Form.

Conclusion

We had made the process lot easier, you are just one click away from booking your seats and view your booking and unbook them. You can book and unbook for yourself or for anyone. This project includes a user-friendly interface for ticket management,with optimized queries and normalized databases. They make the system more efficient to store and organize data than spreadsheets and also allow a centralized facility such that the data can easily be accessed,modified and quickly shared among multiple users.

Also as an admin, you can handle all your seats and buses with the help of just a button. We made use of database in the way to help this process.

Thank You...



Destination arrived