DBMS Project

Online Railway Reservation System

Submitted to :

DR PONMAGAL RS

Submitted By :

1. ILANSURYA (954)

2. ANKITH KUMARAN(961)

Type your text

Contents:

- **Introduction**
- **ER Diagram**
- **Normalisation**
  - First Normal Form
  - Second Normal Form
  - Third Normal Form
- **SQL Queries**
- **SQL Code**

1. INTRODUCTION

This system is basically concerned with the reservation and cancellation of railway tickets to the passenger. The need of this system arouse because as is the known fact that India has the largest railway network in the whole world and it is not possible to handle such a large system manually. By computerizing it, it became possible to overcome the limitations and make the system operations more efficient. The complexity in handling data and records of such a vast system got reduced and became easier by computerizing the system.

Being more specific, this online railway reservation system can perform the basic functions like reservation and cancellation. The users are required to register on the server for getting access to the database and query result retrieval. Upon registration completion, each user has an account which is essentially referred to as the 'view level' of the customer. The account contains comprehensive information of the user entered during the registration and allows the user to access their past reservations, cancellations, enquire about trains and train schedule, seat availability and make afresh reservations. The user will also be able to update their account details, etc.

The master user of this system is the Railway Administrator who can login using a master password and once a user is authenticated as the admin, he/she can access and modify information stored in the database of this system. This includes adding and updating of train, station, train routes and also managing the user and the passenger details.

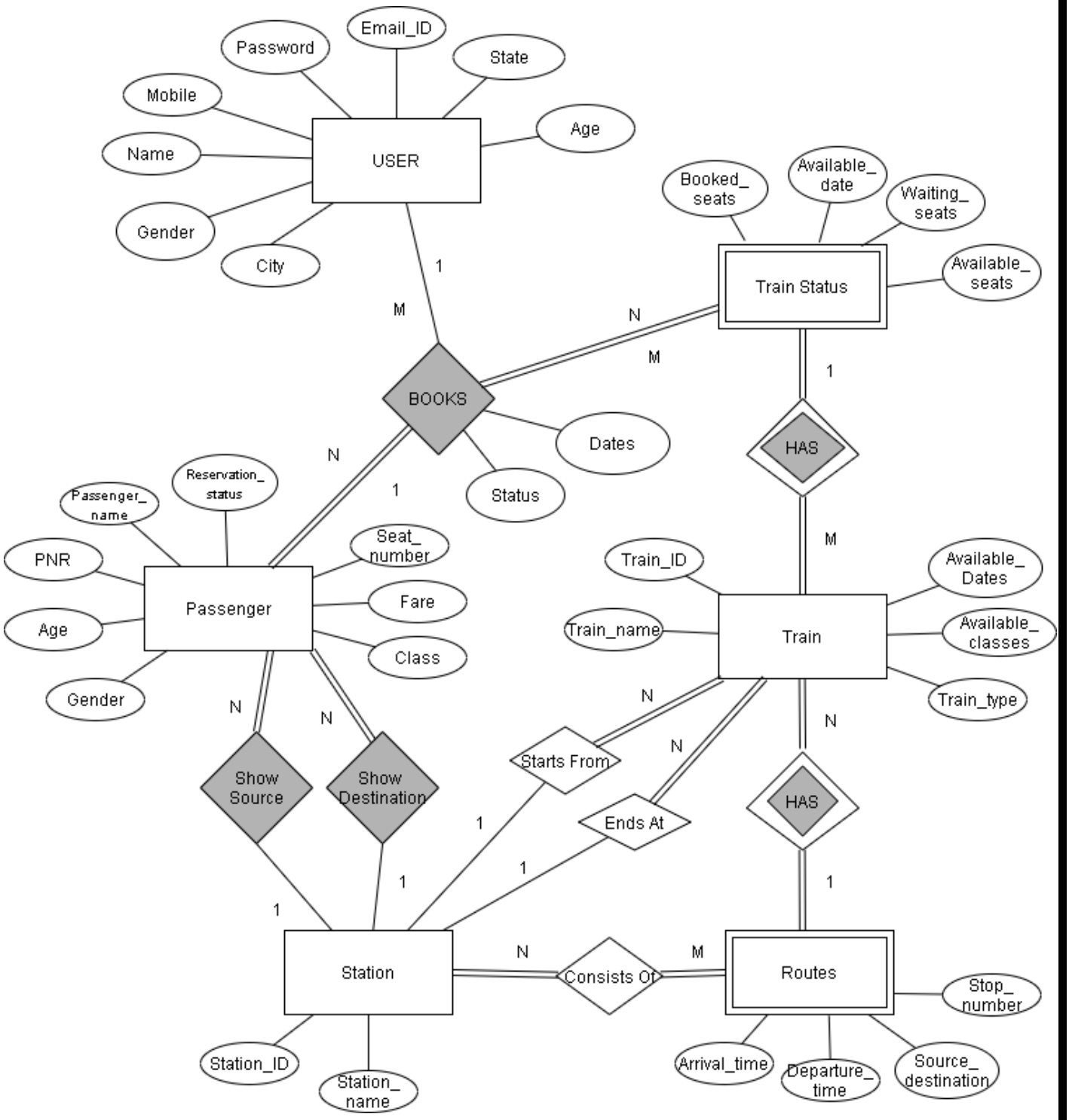
2. ENTITY RELATIONSHIP DIAGRAM

- **Entity Types:**

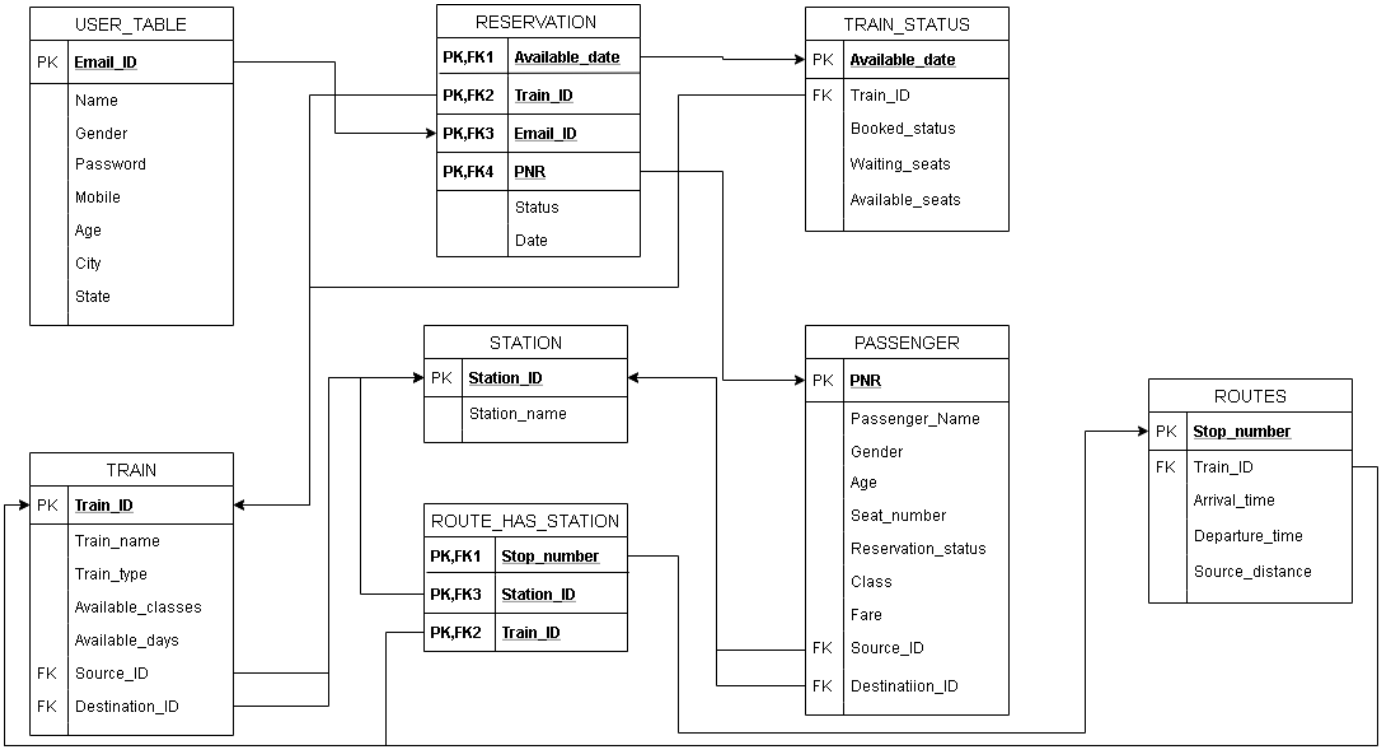
S No.	Entity	Attributes
1.	User	Email id, Password, Name, Gender, Age, City, Mobile, Status.
2.	Passenger	PNR, Passenger Name, Reservation Status, Age, Gender, Seat Number.
3.	Train	Train Id, Train Name, Train Type, Available Classes.
4.	Station	Station Id, Station Name.
5.	Train Status	Booked Seats, Available Seats, Waiting Seats, Available Date.
6.	Route	Arrival Time, Departure Time, Source Distance, Stop Number.

- **Relationships:**

S No.	Relation Types	Entity Types Involved
1.	Consists of	Station, Route
2.	Starts From/Ends on	Train, Station.
3.	Has	Train, Train Status.
4.	Books	Passenger, User, Train Status.
5.	Show Source/Show Destination	Passenger, Station.


Email_ID
Password
State
Mobile
Age
Name
USER
Gender
City
1
M
N
M
Booked_ seats
Available_ date
Waiting_ seats
Train Status
Available_ seats
1
BOOKS
Dates
HAS
N
1
Status
Reservation_ status
Passenger_ name
Seat_ number
PNR
Passenger
Fare
Age
Class
Gender
M
Train_ID
Available_ Dates
Train_name
Train
Available_ classes
Train_type
N
N
N
N
Starts From
Show Source
Show Destination
Ends At
HAS
1
1
1
1
1
Station
N
Consists Of
M
Routes
Stop_ number
Station_ID
Station_ name
Arrival_time
Departure_ time
Source_ destination

Initial Relational schema



3. Normalisation

First Normal Form:

- Relations in 1NF:
  - User
  - Station
  - Train Status
  - Route
  - Route Has Station
- Relations Not in 1NF:
  - Passenger
  - Train

Relations after 1NF:

- Passenger
  This is divided into Passenger and Passenger_Ticket
  - Passenger_Ticket

PNR	Class_Type	Fare	Source ID	Destination ID

  - Passenger

PNR	Seat Number	Passenger Name	Age	Gender

- Train
  This is divided into Train Days and Train
  - Train Days

Train Id	Days Available

  - Train

Train Id	Train Name	Source Id	Train Type	Destination ID

Second Normal Form:

- Relations in 2NF

  - User
  - Station
  - Route Has Station
  - Train Status
  - Route
  - Reservation
  - Passenger
  - Passenger Ticket
  - Train
  - Train Class

All of them are in Second Normal Form.

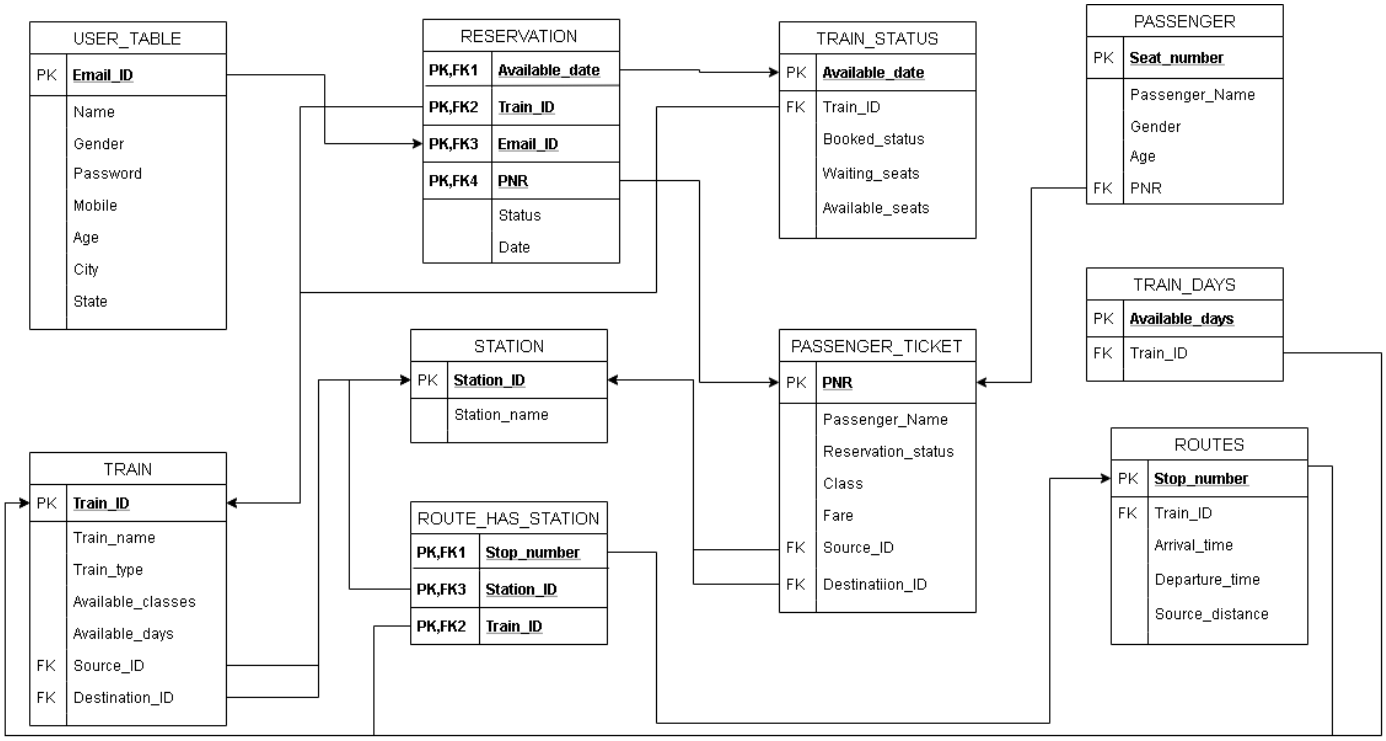
Third Normal Form:

- Relations in 3NF

  - User
  - Station
  - Route Has Station
  - Train Status
  - Route
  - Reservation
  - Passenger
  - Passenger Ticket
  - Train
  - Train Class

All of them are in Third Normal Form.

Relational Shema After Normalisation



4. SQL Queries

- Adding a new Station

```
Insert into Station values (“+ txt_id.Text + “’,’” + txt_name.Text + “’)
```

- Adding a Route

```
Insert into Route values (“ + txt_trainId2.Text + “’,’” + txt_stopno.Text +
“’.’”+txt_stnid.Text + “’,’” + txt_arrival.Text + “’,’” + txt_departure.Text + “’,’”
+ txt_distance.Text + “’)
```

- Read data from Station Table to get Source Station Id

```
Sqlcommad cmd = new sqlcommand(“Select * from Station where Station_name = “’
+ ddl_source.SelectedItem.ToString() + “’”, con1);

SqlDataReader dr = cmd.ExecuteReader();
While (dr.Read())
{
Txt_sid.Text = dr[“Station_Id”].ToString();

}
Dr.Close();
```

- Get seat status from Train_status and display it when user selects the class from dopdownlist provided

```
Sqlcommand cmd = new Sqlcommand("Select * from Train_Id ="" +
lbl_id.Text.ToString() + ""and Available_Date="" + txt_date +""". Con1) ;
    SqlDataReader dr = cmd.ExecuteReader();
    while(dr.Read())
            {
            lbl.seats.Text = dr["Available_seats"].ToString();
            label7.Text = "Total available seats in "+lbl_class.Text + "are : " ;
            lbl_msg.Text = "Total Seats already booked:" ;
            lbl_booked.Text = dr["Booked_seaats"].ToString();

            }
Sr.Close();
```

- When user cancels a ticket, the PNR status is set as Cancelled

```
update Passenger_Ticket set Reservation_status = 'CANCELLED' where PNR =
"+txt_PNR.Text+"
```

5. SQL Code

```sql
DROP DATABASE IF EXISTS RailwayDB;
CREATE DATABASE RailwayDB;
USE RailwayDB;

CREATE TABLE USER_TABLE (
    Email_ID varchar(30) not null PRIMARY KEY,
    Password varchar(10) not null,
    Name varchar(30) not null,
    Gender varchar(8) not null,
    Age INT not null,
    Mobile varchar(10) not null,
    City varchar(20) not null,
    State varchar(30) not null
);

CREATE TABLE STATION (
    Station_ID varchar(10) not null PRIMARY KEY,
    Station_name varchar(30)
);

CREATE TABLE PASSENGER_TICKET (
    PNR varchar(30) not null PRIMARY KEY,
    Class varchar(30) not null,
    Reservation_status varchar(10) not null,
    Source_ID varchar(10) not null,
    Destination_ID varchar(10) not null,
    Fare int not null,
    FOREIGN KEY (Source_ID) REFERENCES STATION (Station_ID),
    FOREIGN KEY (Destination_ID) REFERENCES STATION (Station_ID)
);

CREATE TABLE PASSENGER (
    PNR varchar(30) not null,
    Passenger_name varchar(30) not null,
    Gender varchar(8) not null,
    Seat_number int not null PRIMARY KEY,
    Age int not null,
    FOREIGN KEY(PNR) REFERENCES PASSENGER_TICKET(PNR)
```

```
    FOREIGN KEY(PNR) REFERENCES PASSENGER_TICKET(PNR)
);

CREATE TABLE TRAIN (
    Train_ID INT not null PRIMARY KEY,
    Train_name varchar(30) not null,
    Train_type varchar(20) not null,
    Source_ID varchar(10) not null,
    Destination_ID varchar(10)  not null,
    Available_classes varchar(20) not null,
    FOREIGN KEY (Source_ID) REFERENCES STATION (Station_ID),
    FOREIGN KEY (Destination_ID) REFERENCES STATION (Station_ID)
);

CREATE TABLE DAYS_AVAILABLE (
    Train_ID INT not null,
    Available_days varchar(30) not null PRIMARY KEY,
    FOREIGN KEY (Train_ID) REFERENCES TRAIN (Train_ID)
);

CREATE TABLE TRAIN_STATUS (
    Train_ID int not null,
    Available_date varchar(20) not null PRIMARY KEY,
    Booked_seats int not null,
    Waiting_seats int not null,
    Available_seats int not null,
    FOREIGN KEY (Train_ID) REFERENCES TRAIN (Train_ID)
);

CREATE TABLE ROUTES (
    Train_ID int not null,
    Stop_number int not null PRIMARY KEY,
    Arrival_time text not null,
    Departure_time text not null,
    Source_distance int not null,
    FOREIGN KEY (Train_ID) REFERENCES TRAIN (Train_ID)
```

```
);

CREATE TABLE ROUTES (
    Train_ID int not null,
    Stop_number int not null PRIMARY KEY,
    Arrival_time text not null,
    Departure_time text not null,
    Source_distance int not null,
    FOREIGN KEY (Train_ID) REFERENCES TRAIN (Train_ID)
);

CREATE TABLE RESERVATION (
    PNR varchar(30) not null,
    Train_ID INT not null,
    Email_ID varchar(30) not null,
    Available_date varchar(20) not null,
    Status varchar(20) null,
    CONSTRAINT PK_RESERVATION PRIMARY KEY (Train_ID, Available_date, Email_ID, PNR),
    FOREIGN KEY (Train_ID, Available_date) REFERENCES TRAIN_STATUS (Train_ID, Available_date),
    FOREIGN KEY (Email_ID) REFERENCES USER_TABLE (EMAIL_ID),
    FOREIGN KEY (PNR) REFERENCES PASSENGER_TICKET (PNR)
);

CREATE TABLE ROUTE_HAS_STATION (
    Train_ID INT not null,
    Station_ID varchar(10) not null,
    Stop_number int not null,
    CONSTRAINT PK_ROUTE_HAS_STATION PRIMARY KEY (Train_ID, Stop_number, Station_ID),
    FOREIGN KEY (Train_ID, Stop_number) REFERENCES ROUTES (Train_ID, Stop_number),
    FOREIGN KEY (Station_ID) REFERENCES STATION (Station_ID)
);
```